



**Escuela Politécnica Superior  
Departamento de Ingeniería Informática**

**MÉTODOS CLÁSICOS Y DE SOFT-COMPUTING  
EN LA OPTIMIZACIÓN DE PROCESOS COMPLEJOS.  
APLICACIÓN A UN PROCESO DE FABRICACIÓN.**

**TESIS DOCTORAL**

**Autor:  
Raúl Mario del Toro Matamoros**

**Director:  
Dr. D. Rodolfo E. Haber Guerra**

**Madrid, Febrero de 2011**



**Título de la Tesis:** Métodos clásicos y de soft-computing en la optimización de procesos complejos. Aplicación a un proceso de fabricación.

**Autor:** Raúl Mario del Toro Matamoros

**Director:** Dr. Rodolfo E. Haber Guerra

**Lector:** Prof. Pablo Varona Martínez

Miembros del Tribunal de Defensa de la Tesis Doctoral:

**Presidente:** Prof. José Dorronsoro Ibero

**Vocales:** Prof. José Manuel Andújar Márquez

Prof. Matilde Santos Peña

Dr. José Ramón Alique López

**Secretario:** Dr. Manuel Sánchez-Montañés Isla

**Suplentes:** Dr. Ángel Alique Palomar

Prof. José Eugenio Naranjo Hernández





*A mi madre, Mercedes Matamoros,  
por su educación y la grandeza de su amor hacia mí.*

*A mi abuela Benicia Audinot,  
por su cariño, protección y amor sin límites.*



## AGRADECIMIENTOS

¡Hemos llegado al final del largo túnel, ya podemos acariciar la luz! Aunque ahora comienzan nuevos retos, los de verdad, los que marcarán nuestro paso por esta vida. Hay muchas personas que han incidido directamente en la consecución de este inmenso recorrido, ya sea desde el punto de vista profesional o personal, y que no quiero dejar de mencionar y agradecer en este pequeño momento de gloria.

A Rodolfo, director de esta Tesis, tutor de toda la carrera investigadora y amigo. Siempre dirás que no te debo nada, que el esfuerzo ha sido mío (lo repites con frecuencia), pero debo agradecer enormemente tu paciencia, amistad y, sobre todo: tu confianza desde el principio, tu conducción, tu protección y lealtad frente a todos los obstáculos que ha habido que sortear.

A los colegas (amigos) del equipo investigador de GAMHE y C4LIFE, sin este gran equipo hubiese sido imposible llevar adelante este proyecto. Muchas gracias a Fernando, Alfonso, Maritza (la *amiguita* de la eterna sonrisa) y algunos que ya no están como Pedro, Bruno, Diego y Michael. A Agustín, por su paciencia infinita y por ir siempre despejando el camino. Muchas gracias a Salvador Ros, Ángel Alique, José Ramón Alique, Miguel Ramírez y Jorge Pamies por todo su apoyo. Especial mención a Rodolfo Haber Haber, por sus enseñanzas, confianza e ideas.

Gracias a los colegas del instituto, mejor dicho “intituto”, creo que así tiene más ritmo, que le vamos a hacer, vengo de donde vengo, a la comunidad de “La del viernes”, los que compartimos muchos mediodías, cumpleaños y largos debates. A Miguel y Andrés por su ánimo en las largas tardes, a Gregorio por sus chistes y cafés y a mucha gente del instituto que hacen que el trabajo sea menos difícil.

A los amigos y amigas, no los menciono uno por uno, cuando la amistad es verdadera las palabras sobran. Ustedes saben quienes son, les agradezco estar ahí y la comprensión ante mi escasa comunicación durante todo este tiempo. Para los que han querido emprender también este camino y por circunstancias de la vida han tenido que desistir (Jorge, Héctor, Garzón...) y para los que aún lo intentan en condiciones menos propicias (Guillermo, Elsy, Frank...), este logro también es de ustedes.

A mi madre, a ella debo mi educación y todo lo que soy, a mi abuela Benicia, mi segunda madre, a ustedes dos está dedicado este momento. A mi padre, mis abuelos Julián y Emelina y a toda la gran familia.

Muchas gracias a la tierra que me vio nacer y a mucha de su gente, por la educación básica, la formación y por moldear mi actitud ante los problemas y ante la vida, por enseñarme a sonreír inclusive en los momentos más tristes.

A la familia López-Sánchez, por acogerme como uno más. Mónica, muchas gracias por la paciencia (han sido muchísimas horas sacrificadas), por el empuje, por el amor incondicional, por la compañía, por aportar otros criterios y diferentes puntos de vista, este es: nuestro momento.

## RESUMEN

En esta tesis se presenta el diseño, desarrollo e implementación de sistemas inteligentes, basados en diferentes estrategias de hibridación de técnicas del *soft-computing*, para el control de procesos complejos. La metodología de diseño parte de la obtención de una representación computacional eficiente y simple, en cuanto a implementación algorítmica se refiere, de procesos de elevada complejidad. Estos modelos constituyen la base para el desarrollo de funciones de coste, esenciales durante el procedimiento de ajuste óptimo de los sistemas de control inteligente. Igualmente, estos modelos son básicos para los estudios de simulación a través del empleo de herramientas computacionales de análisis de sistemas dinámicos.

Debido a su gran impacto en la producción de bienes y servicios, las estrategias desarrolladas son aplicadas a la optimización de un proceso de fabricación, en concreto, el proceso de taladrado de alto rendimiento. Pese a su importancia, el conocimiento científico-técnico disponible para su mejoramiento y optimización aún es insuficiente y los resultados contemporáneos son todavía parciales. La incorporación de estrategias de control inteligente a este proceso constituye una alternativa a métodos tradicionales y conduce a un potencial incremento de su productividad lo que, a la larga, se traduce en mayores beneficios económicos.

Las estrategias computacionales diseñadas e implementadas en Matlab, C/C++ y Labview, están dirigidas hacia la optimización del proceso de taladrado de alto rendimiento por medio de técnicas lineales y de *soft-computing*, que constituyen, desde el punto de vista científico y técnico, algunas de las aportaciones de este trabajo. Concretamente, las estrategias propuestas están basadas en un algoritmo de control lineal proporcional, integral y derivativo (PID) y en sistemas híbridos, obtenidos a partir de la combinación de técnicas básicas del *soft-computing*. En particular, se han aprovechado las sinergias entre los sistemas de inferencia borrosa, las redes neuronales adaptativas (sistema híbrido neuro-borroso) y diferentes métodos metaheurísticos tales como los algoritmos genéticos, los métodos basados en la simulación de procesos físicos como el algoritmo de temple simulado (*Simulated Annealing*) y el algoritmo

de entropía cruzada (*Cross Entropy*) y métodos de búsqueda directa como el algoritmo de Nelder-Mead.

Para el ajuste óptimo de las estrategias de control inteligente, se ha utilizado, como función objetivo a minimizar, la integral del error absoluto ponderado en el tiempo (*ITAE*). Esta función penaliza las respuestas transitorias temporales de larga duración y es muy sensible ante variaciones de parámetros, en comparación con otros criterios como la integral del error absoluto (*IAE*) o la integral del error cuadrático ponderado en el tiempo (*ITSE*), entre otros.

Además de las aportaciones expuestas anteriormente, otras contribuciones de esta Tesis Doctoral constituyen los modelos computacionales eficientes que, diseñados e implementados mediante una herramienta de alto rendimiento como Matlab/Simulink, permiten emular el comportamiento dinámico o estático de variables representativas en procesos de fabricación.

Como parte de la metodología científica de la Tesis Doctoral, todas las estrategias desarrolladas son validadas rigurosamente en una plataforma experimental, utilizada como soporte tecnológico, lo cual permite el aprovechamiento mutuo de las experiencias alcanzadas y la mejora de los resultados obtenidos. La validación de las estrategias mediante su implementación en tiempo real, actuando directamente sobre operaciones de taladrado de alto rendimiento, ha permitido corroborar los resultados obtenidos en simulación. Además, ha dado la posibilidad de realizar estudios comparativos entre las estrategias basadas en técnicas de *soft-computing* y las estrategias lineales para comprobar su desempeño, así como la efectividad de los métodos metaheurísticos empleados en el ajuste de los sistemas de control inteligente. Los estudios comparativos se han basado en la cuantificación del cumplimiento de objetivos de control, a partir de parámetros que caracterizan el comportamiento dinámico de sistemas de control, como la amplitud máxima de las oscilaciones (máximo sobrepaso) que pueden aparecer en la variable controlada (ej., fuerza de corte) y los valores medios de su error respecto a su valor deseado (ej., índices de desempeño *ITAE*, *IAE*, *ITSE*, entre otros).

Los estudios comparativos mostraron que, en el caso de la aplicación de las técnicas de optimización para el ajuste de sistemas de inferencia borrosa, las respuestas temporales de los sistemas de control, obtenidas en simulación son similares a las experimentales. Esto se debe a que en todos los casos se han obtenido parámetros óptimos o sub-óptimos de los sistemas de control borrosos. Pero a pesar de su similitud, el desempeño en general del sistema de control optimizado mediante el método de entropía cruzada, es ligeramente superior al resto de técnicas utilizadas, además de ofrecer una muy buena relación tiempo de cómputo y total de parámetros de ajuste del método requeridos.

Por otra parte, los estudios comparativos del desempeño en tiempo real de las estrategias de control no lineal empleadas, basadas en técnicas del *soft-computing*, controlando operaciones de taladrado de alto rendimiento, han corroborado que, en general, resultan más

efectivas que las estrategias lineales para hacer frente a la complejidad intrínseca de estos procesos.

Algunos de los resultados parciales alcanzados durante las diferentes etapas de desarrollo de esta Tesis Doctoral, han sido sometidos a la consideración de la comunidad científica internacional. Investigadores de diferentes campos tales como las ciencias de la computación, la inteligencia artificial, la automática, la ingeniería de fabricación, la ingeniería mecánica y los procesos de transformación de metales, entre otros, han valorado de forma positiva varias de las aportaciones realizadas.





## ABSTRACT

This dissertation presents the design, development and implementation of intelligent systems based on different hybridization strategies of soft-computing techniques for controlling complex processes. The design methodology includes dynamic modelling by obtaining simple and efficient computational representations, in terms of algorithmic implementation, of highly complex processes. These models are the basis for the development of cost functions which are key issues for the optimal tuning procedure of intelligent control systems. Moreover, these models are fundamentals for simulation studies through the use of computational tools for dynamic systems analysis.

Due to its large impact on the goods and services production, the developed strategies are applied to the optimization of a manufacturing process, in particular, the high performance drilling process. Despite its importance, the scientific and technical knowledge generated about its optimization are insufficient and the existing results are still partials. The introduction of intelligent control strategies is an alternative to deal with its physical complexity as well as to yield a potential increase in productivity that stem on greater economic benefits.

The computational control strategies designed and implemented in Matlab, C/C++ and Labview are directed towards the optimization of the high performance drilling process. From the scientific and technical standpoint, the developed strategies are contributions of this work. More specifically, the strategies are based on the proportional, integral and derivative (PID) control algorithm as well as hybrid intelligent systems, generated from the combination of basic soft-computing techniques. In particular, it has taken advantage from the synergies between fuzzy inference systems, adaptive neural networks (neuro-fuzzy hybrid system) and diverse metaheuristics methods. Methods such as the genetic algorithms and based on simulation of physical processes (i.e., simulated annealing algorithm and the cross entropy algorithm). Direct search methods such as the Nelder-Mead algorithm are also considered.

For the optimal tuning of the intelligent control strategies, the integral of the time weighted absolute error (*ITAE*) has been used as objective function to minimize. This function

penalizes the long time duration transient responses. Likewise, it is very sensitive to changes in control system parameters, compared with other criteria such as integral absolute error (*IAE*) or the integral of the time weighted square error (*ITSE*), among others.

In addition of the contributions discussed above, the design and implementation of efficient computational models constitute other valuable contributions of this Thesis. These models emulate the dynamic or the static relations of the most representative variables in manufacturing processes. Being possible the implementation due to the use of a high performance tool for the technical computing as Matlab/Simulink.

As part of the scientific methodology of this Thesis, the developed control strategies are rigorously validated in an experimental platform used as technological support. This method enables a mutual advantage of the acquired experience and a results improvement. The real time control of experimental high performance drilling operations corroborates the results obtained during the simulation stage. Moreover, these experiments give the possibility to make comparative studies between the control strategies. Also, they allow comparing the effectiveness of metaheuristics methods employed for optimal tuning of intelligent control systems. The comparative studies are based on performance indices that characterize the dynamic behaviour of the control systems. Performance indices related to the controlled variable (i.e., cutting force) such as the maximum overshoot (i.e., maximum amplitude of oscillations) and average values of its error with respect to the reference value (i.e., *ITAE*, *IAE*, *ITSE*, among others).

Comparative studies confirm that the fuzzy control systems optimally tuned by different methods, provide a satisfactory performance of the process. The dynamic transient responses of control systems obtained in simulation and experimentally, are very similar. The cause is that in all cases the adjusted controllers parameters are optimal or sub-optimal. But despite their similarity, the overall performance of the control system optimized by the cross-entropy method is slightly superior to the others. It also offers a good relation between the computational time and total setting parameters that requires.

Moreover, comparative studies have confirmed that the nonlinear control strategies based on soft-computing techniques, in general, have a better performance than linear strategies for real-time control of high performance drilling operations. Therefore, are more effective to deal with the inherent complexity of these types of processes.

During the development stages of this Thesis, some of the partial achieved results have been submitted for consideration by the international scientific community. Researchers from different fields such as computer science, artificial intelligence, automation, manufacturing engineering, mechanical engineering and metallurgical engineering, among others, have positively evaluated some contributions of the reported work.

# TABLA DE CONTENIDO

LISTA DE FIGURAS .....	XV
LISTA DE TABLAS .....	XIX
INTRODUCCIÓN .....	1
Objetivos .....	3
Metodología.....	5
Estructura del trabajo .....	5
CAPÍTULO 1 ESTADO DEL ARTE.....	9
1.1    Sistemas Inteligentes para el control y monitorización de procesos .....	9
1.2    Operaciones de mecanizado como procesos electro-mecánicos complejos .....	19
1.3    Inteligencia Artificial para la optimización de los procesos de mecanizado .....	22
1.4    Conclusiones.....	27
CAPÍTULO 2 UN ENFOQUE COMPUTACIONAL PARA EL MODELADO DE PROCESOS DE FABRICACIÓN.....	29
2.1    Identificación de sistemas dinámicos .....	30
2.1.1    Pre-procesamiento de los datos para la identificación experimental .....	33
2.1.2    Plataforma experimental para la identificación de procesos de fabricación .....	35
2.2    Diseño e implementación de un modelo computacional para emular el comportamiento de variables representativas en procesos de fabricación. Análisis del caso particular de las fuerzas de corte en el proceso de fresado de alta velocidad .....	38
2.2.1    Modelo físico-matemático del proceso de fresado de alta velocidad .....	39
2.2.2    Descripción del modelo implementado en Simulink.....	46
2.2.3    Validación experimental del modelo de fuerzas de corte .....	52
2.3    Diseño e implementación de un modelo dinámico computacionalmente eficiente de un proceso de fabricación complejo: el proceso de taladrado de alto rendimiento.....	54
2.3.1    Modelado dinámico del proceso de taladrado.....	56
2.3.2    Identificación experimental del proceso TAR en una máquina herramienta .....	59
2.3.3    Implementación de los modelos en Matlab/Simulink .....	63
2.3.4    Validación de los modelos y análisis de error de ajuste .....	64

2.3.5	Aplicación de los modelos a la predicción de las fuerzas de corte en materiales de mayor dureza. Análisis de errores .....	66
2.4	Diseño e implementación de un modelo dinámico múltiple entrada-simple salida, computacionalmente eficiente de un proceso de fabricación complejo. Estudio del caso particular del proceso de fresado de alta velocidad .....	68
2.4.1	Diseño de experimento para la identificación del modelo dinámico del FAV .....	69
2.4.2	Identificación experimental del modelo dinámico del proceso de fresado de alta velocidad .....	71
2.4.3	Implementación del modelo en Matlab/Simulink .....	79
2.5	Conclusiones.....	82
CAPÍTULO 3 ESTUDIO DE MÉTODOS EFICIENTES LINEALES, BORROSOS Y NEURO-BORROSOS PARA EL CONTROL DE PROCESOS COMPLEJOS .....		85
3.1	Estudio de estrategias lineales de control (algoritmo PID) y su aplicabilidad al control de procesos complejos.....	86
3.1.1	Métodos de ajuste.....	88
3.1.2	Algunas limitaciones .....	89
3.2	Estudio de los sistemas borrosos y su aplicabilidad al control de procesos complejos.....	89
3.2.1	Algunas definiciones básicas .....	91
3.2.2	Operaciones básicas de la lógica borrosa .....	93
3.2.3	Producto Cartesiano .....	95
3.2.4	Composición “Sup-*” .....	96
3.2.5	Variable lingüística. Definición .....	96
3.2.6	El dispositivo lógico borroso: premisas de diseño .....	97
3.2.7	Configuración básica del DLB.....	99
3.2.8	Características generales del DLB .....	102
3.2.9	La toma de decisiones .....	108
3.2.10	<i>Desemborronado</i> .....	111
3.3	Estudio de los sistemas híbridos neuro-borrosos y su empleo en el control de sistemas complejos .....	113
3.3.1	Redes neuronales artificiales adaptativas .....	115
3.3.2	Arquitectura ANFIS .....	120
3.4	Conclusiones.....	123
CAPÍTULO 4 ESTUDIO DE MÉTODOS DE OPTIMIZACIÓN DE INSPIRACIÓN GEOMÉTRICA, FÍSICA Y BIOLÓGICA EN EL CONTROL DE PROCESOS COMPLEJOS .....		125
4.1	Aplicación de algoritmos de optimización al ajuste de sistemas de control de procesos complejos .....	126
4.2	Estudio del método de optimización Nelder-Mead para su aplicabilidad al ajuste de sistemas de control de procesos complejos .....	129
4.3	Estudio del algoritmo de optimización temple simulado para su empleo en el ajuste de sistemas de control de procesos complejos .....	133
4.4	Breve introducción al estudio de algoritmos genéticos.....	136
4.5	Estudio del método de optimización de entropía cruzada .....	141

4.5.1	Algoritmo de entropía cruzada adaptado para la solución de problemas de optimización.....	143
4.6	Conclusiones.....	145
CAPÍTULO 5 DISEÑO DE SISTEMAS INTELIGENTES DE CONTROL ÓPTIMOS PARA PROCESOS COMPLEJOS .....		147
5.1	Diseño e implementación de una estrategia lineal para la optimización de un proceso de fabricación. Aplicación al caso particular del proceso de taladrado de alto rendimiento.....	148
5.1.1	Simulación del sistema de control.....	149
5.1.2	Implementación experimental del sistema de control .....	151
5.2	Diseño e implementación de un sistema de inferencia neuro-borrosa para el control de un proceso de fabricación.....	154
5.2.1	Modelos directo e inverso.....	155
5.2.2	Simulación del sistema de control.....	159
5.2.3	Implementación experimental del sistema de control y estudio comparativo .....	162
5.3	Diseño e implementación de un sistema de inferencia borrosa aplicado al control del proceso de taladrado de alto rendimiento. Optimización mediante el método de Nelder-Mead.....	164
5.3.1	Diseño de un controlador borroso para el proceso TAR .....	165
5.3.2	Ajuste e implementación del controlador borroso .....	168
5.3.3	Implementación experimental del sistema de control y estudio comparativo .....	174
5.3.4	Estabilidad del sistema de control.....	176
5.4	Control de un proceso de fabricación mediante sistema de inferencia borrosa optimizado a través del método de temple simulado .....	177
5.4.1	Optimización y simulación del controlador borroso .....	178
5.4.2	Resultados experimentales.....	182
5.5	Optimización de un sistema de inferencia borrosa mediante algoritmos genéticos para el control de un proceso de fabricación.....	183
5.6	Ajuste de un sistema de inferencia borrosa, mediante el método de entropía cruzada, para el control de un proceso de fabricación .....	187
5.6.1	Resultados de la simulación.....	188
5.6.2	Resultados experimentales.....	191
5.7	Estudio comparativo entre las técnicas de optimización empleadas para el ajuste de estrategias borrosas de control de un proceso de fabricación complejo.....	192
5.8	Conclusiones.....	196
CONCLUSIONES.....		199
Lista de publicaciones .....		207
Trabajo futuro.....		208
BIBLIOGRAFÍA .....		211



## LISTA DE FIGURAS

Figura 1.1. Principales metodologías del <i>soft-computing</i> iniciales.....	11
Figura 1.2. Componentes del <i>soft-computing</i> y sistemas inteligentes híbridos. ....	12
Figura 1.3. Sistema híbrido neuro-borroso para el control de procesos. ....	15
Figura 1.4. DLB de tipo TSK de primer orden y arquitectura ANFIS equivalente. ....	16
Figura 1.5. Clasificación de los procesos de mecanizado. ....	19
Figura 1.6. Ejemplos de operaciones de mecanizado convencional. ....	20
Figura 1.7. Ejemplos de operaciones de mecanizado no tradicional.....	21
Figura 2.1. Procedimiento de identificación.....	31
Figura 2.2. Respuesta a escalón de un proceso no realimentado. ....	31
Figura 2.3. Algoritmo de filtrado con fase cero.....	34
Figura 2.4. Componentes de la plataforma experimental.....	35
Figura 2.5. Diagrama de la plataforma experimental.....	36
Figura 2.6. Niveles de red en la plataforma experimental de monitorización y control.....	37
Figura 2.7. Esquema del proceso de FAV.....	40
Figura 2.8. Diagrama del proceso de FAV visto como una caja negra. ....	40
Figura 2.9. Modelo geométrico generalizado de una fresa.....	42
Figura 2.10. Geometría general de la herramienta.....	43
Figura 2.11. Operación de fresado en 3D con el ángulo de elevación de la superficie $\Phi$ .....	45
Figura 2.12. Diagrama de bloques del modelo matemático de las fuerzas y el par de corte en SIMULINK.....	47
Figura 2.13. Interfaz de usuario del modelo en SIMULINK para la entrada de parámetros de la geometría de la herramienta.....	48
Figura 2.14. Árbol de dependencias entre los diagramas o subsistemas que conforman el modelo desarrollado. ....	49
Figura 2.15. Otros diagramas de bloques que componen el modelo implementado en SIMULINK.....	51
Figura 2.16. Gráfica de fuerza de corte resultante obtenida del modelo implementado en SIMULINK para ciertos parámetros de entrada. ....	52

Figura 2.17. Fuerza de corte resultante experimental (línea continua) y obtenida por el modelo para diferentes condiciones de corte (línea discontinua).....	53
Figura 2.18. Esquema del proceso de taladrado de alto rendimiento.....	55
Figura 2.19. Parámetros geométricos utilizados en el modelo de Elhachimi et ál. ....	57
Figura 2.20. Diagrama de flujo de la aproximación propuesta por Pirtini y Lazoglu.....	58
Figura 2.21. Diagrama de bloques teórico del proceso de taladrado de alto rendimiento.....	60
Figura 2.22. Perfil para la operación de taladrado. ....	61
Figura 2.23. Señal de entrada al proceso (velocidad de avance) y fuerza obtenida, bruta y media (filtrada) de una operación de taladrado en el material GGG40.....	62
Figura 2.24. Implementación de los modelos en Matlab/Simulink. ....	63
Figura 2.25. Respuesta de los modelos de primer, segundo y tercer orden de la fuerza ante cambios en la velocidad de avance y conjunto de datos de validación. ....	64
Figura 2.26. Función de auto-correlación del error en la predicción de fuerza en el taladrado de GGG40. ....	66
Figura 2.27. Aplicación de los modelos del GGG40 con ganancia unidad a la fuerza normalizada del taladrado del material 17-4PH.....	67
Figura 2.28. Esquema del proceso de FAV y diagrama de bloques de una representación del proceso mediante funciones de transferencia. ....	69
Figura 2.29. Probeta con perfil escalonado en su superficie con cambios de 0.5 mm. ....	69
Figura 2.30. Diseño de experimento para la identificación del proceso de FAV. Cambios planificados para cada una de las variables de entrada al proceso.....	70
Figura 2.31. Señales registradas durante la operación de fresado con la herramienta de 12 mm. ....	72
Figura 2.32. Fuerza media en cada uno de los experimentos. ....	73
Figura 2.33. Funciones de ajuste para las ganancias de los sub-sistemas que componen el modelo.....	75
Figura 2.34. Desarrollo del modelo dinámico del proceso de FAV en Matlab/Simulink. ....	80
Figura 2.35. Simulación del modelo y comparación con los datos experimentales.....	81
Figura 3.1. Diagrama general de un controlador PID en un sistema de control realimentado. ....	86
Figura 3.2. Diagrama en Matlab/Simulink del controlador PID discreto.....	88
Figura 3.3. Sub-conjunto clásico y borroso.....	92
Figura 3.4. Conjunto borroso puntual o singularidad borrosa.....	93
Figura 3.5. Diagramas de Venn. ....	95
Figura 3.6. Variable lingüística fuerza.....	97
Figura 3.7. Diagrama en bloques del lazo de control con DLB.....	99
Figura 3.8. Representación en capas de la red neuronal adaptativa.....	115
Figura 3.9. Arquitectura del sistema ANFIS.....	120
Figura 3.10. Ejemplo de función de distribución de tipo campana generalizada. ....	121
Figura 4.1. Diagrama general de un sistema de control de un proceso físico. ....	126



Figura 4.2. Diagrama de bloques del sistema de control y algoritmo de optimización.....	129
Figura 4.3. Diagrama de flujo del algoritmo de Nelder-Mead. ....	132
Figura 4.4. Algoritmo de Metropolis de aceptación-rechazo. ....	136
Figura 4.5. Diagrama de flujo de datos del algoritmo.....	137
Figura 5.1. Diagrama de simulación del sistema de control de la fuerza de corte.....	149
Figura 5.2. Gráfica del lugar de las raíces para una respuesta aperiódica. ....	150
Figura 5.3. Simulación del sistema de control PID a lazo cerrado. ....	151
Figura 5.4. Diagrama del controlador PID implementado. ....	152
Figura 5.5. Comportamiento en tiempo real del sistema de control. ....	153
Figura 5.6. Estrategia CMI y modelos ANFIS. ....	154
Figura 5.7. Entradas y salidas de los modelos ANFIS. ....	155
Figura 5.8. Funciones de pertenencia y reglas de los modelos ANFIS.....	157
Figura 5.9. Respuesta de los modelos ante datos de validación. ....	158
Figura 5.10. Aplicación de simulación de la estrategia de control por modelo interno y modelos neuro-borrosos. ....	160
Figura 5.11. Simulación de la respuesta del sistema de control ante un escalón en la referencia .....	161
Figura 5.12. Comportamiento del criterio <i>ITAE</i> en presencia de retardos de tiempo. ....	161
Figura 5.13. Comportamiento del sistema de control CMI-ANFIS y otras estrategias de control .....	162
Figura 5.14. Comportamiento del error relativo para las estrategias de control.....	164
Figura 5.15. Particiones borrosas y funciones de pertenencia de las entradas y salidas. ....	166
Figura 5.16. Característica estática del controlador borroso. ....	167
Figura 5.17. Sistema de control distribuido basado en un regulador borroso para el control del proceso de TAR .....	169
Figura 5.18. Aplicación de simulación del sistema de control borroso.....	170
Figura 5.19. Resultados de la simulación del sistema de control. ....	171
Figura 5.20. Comportamiento del sistema de control según varía el tiempo de retardo. ....	172
Figura 5.21. Comportamiento de los índices de desempeño <i>ITAE</i> e $IT^2SE$ según varía el tiempo de retardo. ....	173
Figura 5.22. Comportamiento del proceso según las estrategias de control PID y borrosas.....	174
Figura 5.23. Comportamiento del error absoluto ponderado en el tiempo de las estrategias de control.....	176
Figura 5.24. Optimización mediante temple simulado.....	178
Figura 5.25. Resultados tras 75 ejecuciones del algoritmo de SA. ....	179
Figura 5.26. Simulación del sistema de control borroso optimizado mediante SA. ....	180
Figura 5.27. Comportamiento de la fuerza de corte según la variación del tiempo de retardo.....	181
Figura 5.28. Comportamiento del criterio <i>ITAE</i> según varía el tiempo de retardo. ....	181

Figura 5.29. Respuesta temporal del sistema de control optimizado mediante SA durante el taladrado del material GGG40.....	182
Figura 5.30. Comportamiento del sistema de control optimizado durante una operación de taladrado del material 17-4PH.....	183
Figura 5.31. Diagrama del sistema de control borroso y optimización mediante algoritmos genéticos.....	184
Figura 5.32. Resultados tras 99 ejecuciones de la optimización mediante GA.....	185
Figura 5.33. Simulación del sistema de control optimizado mediante GA.....	186
Figura 5.34. Aplicación de simulación del sistema de control con ruido aditivo.....	186
Figura 5.35. Diagrama del sistema de control borroso y el método de optimización de entropía cruzada.....	188
Figura 5.36. Funciones de densidad probabilísticas para los parámetros $[K_e \ K_{ce}]$ .....	189
Figura 5.37. Resultados durante la optimización mediante entropía cruzada.....	190
Figura 5.38. Simulación del sistema de control borroso.....	190
Figura 5.39. Comportamiento de los criterios de desempeño <i>ITAE</i> e <i>ISTSE</i> en presencia de retardos de tiempo en el sistema de control.....	191
Figura 5.40. Comportamiento del sistema de control durante el taladrado del material GGG40.....	192
Figura 5.41. Estudio comparativo sin presencia de perturbaciones.....	194
Figura 5.42. Estudio comparativo en presencia de perturbaciones.....	196

## LISTA DE TABLAS

Tabla 1.1. Aplicación del <i>soft-computing</i> al modelado, monitorización y control de procesos. ....	18
Tabla 1.2. Técnicas de IA en la optimización de procesos de mecanizado. ....	24
Tabla 1.3. Sistemas inteligentes en la automatización de procesos de mecanizado. ....	25
Tabla 2.1. Comparación de las fuerzas de corte obtenidas experimentalmente y mediante el modelo. ....	54
Tabla 2.2. Resultados del análisis estadístico de los retardos de tiempo del bus de campo MPI. ....	59
Tabla 2.3. Propiedades de los materiales de las piezas de trabajo experimentales. ....	60
Tabla 2.4. Condiciones nominales para la operación de taladrado. ....	61
Tabla 2.5. Modelos del taladrado de GGG40. ....	62
Tabla 2.6. Resultados del análisis de error para los modelos del taladrado de GGG40. ....	65
Tabla 2.7. Análisis del error de ajuste para los modelos generados del material GGG40 empleados en el material 17-4PH. ....	67
Tabla 2.8. Valores que alcanzan cada una de las variables de entrada a lo largo del eje $x$ de la pieza, durante la operación de ranurado. ....	71
Tabla 2.9. Condiciones nominales para el experimento de identificación. ....	71
Tabla 2.10. Parámetros de algunos de los modelos obtenidos con la herramienta de 12 mm. ....	74
Tabla 2.11. Funciones de ajuste de las ganancias de los sub-sistemas para la herramienta de 10 mm. ....	76
Tabla 2.12. Análisis estadísticos de las constantes de tiempo de los modelos de los sub-sistemas. ....	76
Tabla 2.13. Valores de ajuste de los coeficientes para el cálculo de las ganancias del modelo. ....	78
Tabla 2.14. Valores de ajuste de las constantes de tiempo y retardos de tiempo de los sub-sistemas que conforman el modelo del proceso de FAV. ....	78
Tabla 2.15. Errores del modelo, réplicas de ajuste y de validación (filas sombreadas). ....	82
Tabla 3.1. Métodos de ajuste de algoritmos de control PID. ....	88
Tabla 3.2. Algunas funciones de pertenencia típicas. ....	104
Tabla 4.1. Algunos tipos de representación de los individuos de una población. ....	138

Tabla 4.2. Ejemplos de operadores de cruce. ....	139
Tabla 5.1. Criterios de error de la respuesta temporal de la fuerza de corte. ....	153
Tabla 5.2. Resumen de los parámetros de los modelos neuro-borrosos.....	159
Tabla 5.3. Estudio comparativo entre el controlador CMI-ANFIS y otras estrategias de control analizadas. ....	163
Tabla 5.4. Base de reglas del controlador borroso.....	167
Tabla 5.5. Comportamientos extremos del sistema de control para el tiempo de retardo variable. ....	173
Tabla 5.6. Estudio comparativo de las estrategias de control analizadas.....	175
Tabla 5.7. Estudio comparativo de los resultados experimentales y de simulación del comportamiento del controlador borroso optimizado mediante el método de entropía cruzada.....	192
Tabla 5.8. Parámetros de las estrategias de optimización para un ajuste óptimo de un controlador borroso de 49 reglas.....	193
Tabla 5.9. Estudio comparativo del desempeño del sistema de control borroso sin presencia de perturbaciones.....	194
Tabla 5.10. Estudio comparativo del desempeño del sistema de control borroso en presencia de perturbaciones. ....	195

---

# INTRODUCCIÓN

---

Los sistemas físicos involucrados en la industria del siglo XXI forman parte de un sistema social y técnico dinámico en un entorno cada vez más globalizado y turbulento. Los cambios que ha sufrido la economía mundial en las últimas dos décadas, debido al crecimiento económico de nuevos países emergentes, además de fenómenos como la deslocalización, han puesto en el punto de mira a varios sectores considerados estratégicos, entre ellos un sector muy importante y sensible para el empleo, el bienestar social y la seguridad de las naciones: el sector de la fabricación. Los procesos de fabricación tienen, por su variedad y complejidad, un gran impacto en el producto nacional bruto de Europa. Se estima que la actividad de fabricación representa aproximadamente el 22% del PNB europeo, además, un 75% del PIB y un 70% del empleo a nivel europeo están relacionados con la fabricación [1].

Existe la necesidad de desarrollar e implementar estrategias de fabricación basadas en la investigación y la innovación, que aceleren la transformación de la industria europea, asegurando un empleo de valor añadido y logrando que una parte importante de la producción esté soportada en la economía del conocimiento. Esta estrategia debe garantizar una continua competitividad de la industria de fabricación, significando una radical transformación de la industria en todos sus niveles, desde la red de sistemas de fabricación hasta las fábricas, los sistemas de producción, máquinas, componentes y procesos técnicos.

Entre las áreas prioritarias que asegurarán el desarrollo del sector de la fabricación en Europa en los próximos años, se encuentra la relacionada con la transformación de la fabricación de componentes basada en recursos (humanos, computacionales, maquinaria, etc.) en una fabricación basada en conocimiento. Esa transformación va indisolublemente ligada a las Ciencias de las Computación y a la Inteligencia Artificial de modo que permitan una materialización pragmática de la fabricación adaptativa, digital y basada en conocimiento. Es decir, hay que incorporar una respuesta automática a los cambios del entorno, involucrando software y tecnologías convergentes en el diseño y operación de los procesos, además de la

utilización del conocimiento para optimizar los procesos de fabricación, su adquisición y transferencia.

La mejora de la respuesta de los sistemas de fabricación, ante los cambios continuos y permanente evolución del entorno, técnicamente es viable a través del desarrollo de sistemas de control con capacidades de auto-aprendizaje, cooperativos y auto-optimizables. Este hecho implica, ineludiblemente, la incorporación de cierta inteligencia para la mejora de los procesos, lo que a su vez conduce al desarrollo de sistemas de vigilancia y monitorización de bajo coste que mejoran la capacidad de pronóstico, la fiabilidad y el rendimiento de los sistemas de monitorización actuales. A medio plazo la mayor prioridad es el desarrollo de técnicas basadas en la Inteligencia Artificial que permitan una fabricación más eficiente y rápida a través de un comportamiento cooperativo, auto-organizado y auto-optimizado de los sistemas de control de procesos.

Los esfuerzos de inversión económica en la industria, en nuevos sistemas inteligentes de supervisión y control, solo representan un 10% del total a invertir para perfeccionar el sistema productivo, ofreciendo por contraparte un alto valor añadido que mejora ostensiblemente la competitividad en un plazo de tiempo reducido. Es por ello que los sistemas inteligentes de monitorización y control de los procesos industriales, se han convertido en los últimos años en una de las áreas de mayor interés donde la comunidad científica ha centrado buena parte de sus esfuerzos.

Esta Tesis Doctoral se enmarca en la actividad investigadora de dos grupos de reconocido prestigio. El grupo de investigación GAMHE (Grupo de Automatización Avanzada de los Procesos de Mecanizado de Alto Rendimiento), perteneciente al Consejo Superior de Investigaciones Científicas, tiene entre sus líneas principales de investigación, el empleo de sistemas inteligentes para la monitorización y el control de procesos complejos y la optimización de procesos de fabricación, en específico los procesos de mecanizado de alto rendimiento. El grupo C4LIFE de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, tiene como objetivo principal el desarrollo y aplicación de nuevos métodos científico-técnicos y estrategias avanzadas en el marco de la conjunción de la computación y el control inteligente, los sistemas cognitivos artificiales y los componentes basados en software de libre distribución y en tecnologías emergentes de comunicaciones, con vistas a generar nuevos conocimientos científicos y técnicos y optimizar el comportamiento de sistemas complejos de modo que conduzcan a corto, medio y largo plazo a una mejora en la calidad de vida.

En general, la implementación con éxito de los sistemas de monitorización y control depende de tres partes esenciales: la robustez del sistema sensorial, la calidad y cantidad de información de las señales medidas y las técnicas utilizadas para procesar la información y llevar a cabo las tareas de supervisión y control. Cada una de estas áreas ha sido un nicho para

las investigaciones relacionadas con el desarrollo de nuevos sensores, modernos sistemas de procesamiento y estrategias de supervisión y control basadas en técnicas de Inteligencia Artificial (IA).

Desde el punto de vista científico-tecnológico, la complejidad y variedad de los procesos de fabricación, la necesidad de compatibilizar los sistemas de supervisión con los sistemas convencionales en funcionamiento y la necesidad de optimizar el proceso de producción, han sido catalizadores en la aplicación de las llamadas técnicas IA. Un ejemplo de ellos constituye el gran número de aplicaciones a procesos físicos basadas en la Lógica Borrosa, evidenciando el empuje de esta técnica. De hecho, las estrategias de control borrosas son consideradas como un camino idóneo para hacer frente al comportamiento no lineal y variable en el tiempo de los procesos reales, además de la imprecisión y fiabilidad de los sistemas de medición. Otras de las técnicas a destacar son las Redes Neuronales Artificiales que tienen una extraordinaria capacidad para el aprendizaje, la auto-organización de la información, el cómputo paralelo y su viabilidad para la implementación de hardware a la medida y aplicaciones en tiempo real.

Los procesos de mecanizado de alto rendimiento abarcan procesos de corte desde la macro escala hasta procesos de corte en la micro y nano escala, que se consideran de gran complejidad y en los cuales las técnicas de control que se están aplicando ofrecen prestaciones insuficientes o, en algunos casos, prácticamente inexistentes. Por ello, la comunidad científica trabaja de forma activa en el desarrollo de sistemas inteligentes de fabricación y en estrategias de control para optimizar estos procesos. Ciertamente los sistemas de control abierto, y en concreto los Controles Numéricos por Computador (CNC) abiertos, permiten el desarrollo e implementación de sistemas de control y supervisión.

Los trabajos relacionados con el control para optimizar el proceso de corte en el mecanizado convencional abundan en la literatura, sin embargo, y a partir de aquella a la que se ha tenido acceso, hay muy pocos trabajos dirigidos hacia la optimización de los procesos de mecanizado de alto rendimiento y los resultados son todavía parciales. Con esta Tesis Doctoral se ha pretendido generar nuevos conocimientos científicos y técnicos sobre este tema, así como contrastar experimentalmente las aportaciones en el campo de la optimización de determinados procesos de fabricación, proponiendo una metodología susceptible de ser aplicada a otros procesos.

## **Objetivos**

El objetivo principal de este trabajo es el diseño, desarrollo y aplicación de estrategias de optimización basadas en *soft-computing*, en el contexto de los sistemas de control cognitivos artificiales, para el control de procesos complejos [2]. Asimismo, se trata de obtener una representación computacional eficiente y simple, de procesos de elevada complejidad, como

base para el desarrollo de funciones de coste esenciales en los métodos de optimización, ya sean mediante técnicas deterministas ó estocásticas, de inspiración física ó biológica.

Las estrategias desarrolladas serán aplicadas a la optimización de un proceso de fabricación, en específico el proceso de taladrado de alto rendimiento. Este proceso ha sido escogido por su gran impacto en la producción de bienes y en los servicios. A pesar de su importancia, resulta insuficiente el conocimiento científico técnico disponible relacionado con la optimización de este proceso y los resultados existentes son todavía parciales.

El planteamiento de este objetivo tiene asociado los siguientes **retos científicos y técnicos** parciales:

1. Diseñar e implementar un modelo computacional para emular el comportamiento de variables representativas en procesos de fabricación. Análisis del caso particular de las fuerzas de corte en el proceso de fresado de alta velocidad.
2. Diseñar e implementar de un modelo dinámico computacionalmente eficiente de un proceso de fabricación complejo. Estudio del caso particular del proceso de taladrado de alto rendimiento.
3. Diseñar e implementar un modelo dinámico múltiple entrada-simple salida, computacionalmente eficiente, de un proceso de fabricación complejo. Estudio del caso particular del proceso de fresado de alta velocidad.
4. Diseñar e implementar una estrategia lineal para la optimización, en tiempo real, de un proceso de fabricación. Aplicación al caso particular del proceso de taladrado de alto rendimiento.
5. Diseño e implementación de una estrategia de control neuro-borroso para la optimización en tiempo real de un proceso de fabricación complejo.
6. Diseñar e implementar estrategias de optimización, clásicas y contemporáneas, de inspiración física y biológica (i.e., algoritmo heurístico de Nelder-Mead, algoritmos genéticos, entropía cruzada y temple simulado) de un sistema de control de inferencia borrosa de un proceso de fabricación complejo.
7. Desarrollar un estudio comparativo entre técnicas de *soft-computing* y estrategias lineales para el control en tiempo real de un proceso de fabricación complejo.
8. Desarrollar un estudio comparativo entre técnicas de optimización empleadas para el ajuste de estrategias borrosas de control de un proceso de fabricación complejo.

Desde el punto de vista técnico, la solución propuesta en este trabajo tiene como punto de partida el mecanizado inteligente, visto como un procedimiento computacional eficiente desarrollado a partir de la combinación de una o más técnicas de Inteligencia Artificial y criterios de expertos, con uno o más niveles de resolución, que básicamente manipulan las condiciones de operación en los procesos de mecanizado de alto rendimiento, con el objetivo de optimizar estos procesos, ya sea mediante disminución del desgaste en las correspondientes



herramientas de corte, el incremento de la calidad de las piezas fabricadas o el incremento de la productividad en general [3].

También desde el punto de vista técnico, se profundiza en el estudio del desempeño de diferentes estrategias de optimización de algoritmos de control inteligente, de tal forma que la metodología de ajuste sea fácil, intuitiva y susceptible de ser utilizada en aplicaciones de la lógica borrosa al control automático de procesos físicos en general.

## **Metodología**

Para el desarrollo de esta Tesis Doctoral se han utilizado métodos de investigación teóricos y experimentales, interactuando mutuamente. Las estrategias concebidas y los algoritmos desarrollados son fruto de la aplicación de los más adecuados paradigmas elegidos dentro del amplio universo que ponen a nuestra disposición las Ciencias de la Computación y la Inteligencia Artificial, la Teoría de Sistemas y la Automática.

El estudio y selección de las teorías más viables para el modelado, el control y la optimización ha sido precedido por una re-elaboración de estas estrategias y algoritmos que ha permitido su concepción como una doctrina integradora en el diseño de sistemas inteligentes de control óptimos para sistemas electromecánicos complejos.

Las estrategias de control y optimización desarrolladas, han sido objeto de validación experimental rigurosa mediante la utilización de un soporte tecnológico, potenciando de esta forma, la sinergia entre la amalgama de técnicas y procedimientos desarrollados. Esto permite aplicar de manera iterativa, investigación teórica y experimental, pudiendo aprovechar de forma mutua las experiencias alcanzadas y mejorando así, los resultados obtenidos. De este modo se contribuye a una mayor completitud del trabajo de Tesis Doctoral y de los resultados aquí plasmados.

## **Estructura del trabajo**

El trabajo consta de 5 capítulos, además de un acápite de conclusiones donde se resumen los aportes de mayor relevancia y propuestas de líneas de investigación futuras.

En el Capítulo 1 se realiza una revisión del estado de la ciencia y la técnica en los paradigmas elegidos dentro del amplio universo que ponen a nuestra disposición la Teoría General de Sistemas, la Teoría de Control, las técnicas de Inteligencia Artificial y las Ciencias de la Computación. Se presenta, además, una breve revisión de los procesos de fabricación y específicamente los de mecanizado, como caso real de procesos electro-mecánicos complejos, presentando como han evolucionado estos procesos y las tendencias actuales hacia un aprovechamiento efectivo de la complementariedad con otras áreas científicas y tecnológicas.

El Capítulo 2 presenta el diseño e implementación en un lenguaje de alto rendimiento, como Matlab/Simulink, de modelos de procesos de fabricación complejos que son obtenidos a partir de las ecuaciones físico-matemáticas de los procesos, así como también mediante el empleo de técnicas de identificación experimental. En el capítulo se describe la técnica de identificación de sistemas dinámicos empleada, además de las herramientas y algoritmos utilizados durante los procedimientos de identificación, así como también la plataforma experimental empleada para el estudio de procesos de fabricación complejos. Las aportaciones principales de este capítulo son: el diseño e implementación, a partir de su modelo físico, de un modelo computacional para emular el comportamiento de las variables más representativas en procesos de fabricación, en específico el proceso de fresado de alta velocidad; el diseño e implementación de modelos computacionales del comportamiento dinámico de procesos de fabricación complejos, obtenidos mediante técnicas de identificación experimental, en específico modelos simple entrada-simple salida (i.e., proceso de taladrado de alto rendimiento) y múltiple entrada-simple salida (i.e., proceso de fresado de alta velocidad).

El Capítulo 3 recoge un estudio de técnicas clásicas y de *soft-computing* para el control de procesos complejos. Se abordan las ideas esenciales y conceptuales de los sistemas borrosos, los sistemas híbridos neuro-borrosos y de estrategias clásicas, como el algoritmo de acción proporcional, integral y derivativa (PID). Las técnicas estudiadas serán posteriormente utilizadas en el Capítulo 5 para el diseño de sistemas de control óptimos de procesos de fabricación complejos.

En el Capítulo 4 se lleva a cabo un estudio de métodos de optimización de diversas fuentes de inspiración que serán empleados en el Capítulo 5 durante los procedimientos de ajustes de las estrategias de control introducidas en el Capítulo 3. Se introducen los conceptos básicos de los algoritmos de optimización, así como una estrategia para su aplicación al ajuste de sistemas de control. Uno de los algoritmos considerados es el método de optimización Nelder-Mead, que es una técnica de optimización no-lineal que goza de gran popularidad y además no utiliza información sobre el gradiente de los datos, por lo que resulta adecuado para ser aplicado en problemas con funciones discontinuas o con derivadas abruptas. Otro es el algoritmo de optimización temple simulado o recocido simulado (*Simulated Annealing*), que es un método de búsqueda meta-heurística para la resolución de problemas de optimización global y está inspirado en el proceso físico de recocido de metales. También se han considerado los algoritmos genéticos, que están basados en la selección natural y han sido además, ampliamente utilizados en la solución de diversos tipos de problemas de optimización. Adicionalmente se exponen las ideas conceptuales de uno de los métodos de optimización global más novedosos: el algoritmo de entropía cruzada (*Cross Entropy*). Este se enmarca dentro de los métodos de tipo Monte Carlo, permitiendo la optimización de problemas combinatoriales, continuos y con multiplicidad de extremos.

El Capítulo 5 se centra, principalmente, en el diseño de sistemas de control óptimos, basados en técnicas clásicas y de *soft-computing*, de procesos complejos y su aplicación a un proceso de fabricación, en específico el taladrado de alto rendimiento. La metodología de diseño empleada en el capítulo dispone entre sus principales etapas, la simulación de estrategias de control desarrolladas a partir de las técnicas introducidas en el Capítulo 3 y su implementación en la plataforma experimental descrita en el Capítulo 2. El diseño de los sistemas de control se realiza a partir de modelos de los procesos de fabricación complejos aportados en el Capítulo 2, en específico, a partir del modelo dinámico, computacionalmente eficiente, del proceso de taladrado de alto rendimiento. Las principales aportaciones de este capítulo son: el diseño e implementación de una estrategia de control lineal para la optimización de un proceso de fabricación complejo, en específico el proceso de taladrado del alto rendimiento; el diseño e implementación de estrategias de optimización, clásicas y contemporáneas, tanto de inspiración geométrica, física como biológica, para el ajuste de un sistema de inferencia borrosa empleado para el control de un proceso de fabricación complejo; así como un estudio comparativo de los resultados obtenidos de la aplicación de las diferentes estrategias de optimización. Asimismo, se considera otra aportación el diseño e implementación de un sistema de control por modelo interno, basado en modelos híbridos neuro-borrosos, aplicado también al proceso de taladrado de alto rendimiento.



---

# CAPÍTULO 1

## ESTADO DEL ARTE

---

En este capítulo se presenta una breve revisión del estado de la ciencia y la técnica en los paradigmas elegidos dentro del amplio universo que ponen a nuestra disposición la Teoría General de Sistemas, las técnicas de Inteligencia Artificial, las Ciencias de la Computación y la Automática, **considerando específicamente la conjunción de estas grandes áreas de conocimiento**. Se presenta, además, una breve revisión de los procesos de mecanizado, como caso de estudio de procesos electro-mecánicos complejos, presentando cómo han evolucionado estos procesos y las tendencias actuales hacia un aprovechamiento efectivo de la complementariedad con otras áreas científicas y tecnológicas.

El capítulo se divide en 3 secciones. En la primera sección (sección 1.1) se muestra una revisión de las diversas técnicas provenientes, fundamentalmente, de las Ciencias de la Computación y vinculadas con el campo de la Inteligencia Artificial, con vistas a ser aplicadas al modelado, control y monitorización de procesos físicos. Como procesos electro-mecánicos a estudiar y optimizar, en la sección 1.2 se realiza una introducción a los procesos de mecanizado, presentando su evolución y tendencias actuales. Finalmente, en la sección 1.3, se recogen varias de las aportaciones disponibles en la literatura, que demuestran la viabilidad e importancia de la optimización de los procesos de mecanizado mediante la aplicación de técnicas de Inteligencia Artificial.

### 1.1 Sistemas Inteligentes para el control y monitorización de procesos

Los últimos años se han caracterizado por el desarrollo de nuevos paradigmas en el campo de la Inteligencia Artificial (IA). El uso masivo de las técnicas de IA en el modelado, la monitorización y el control de procesos, ha sido consecuencia de que, en buena medida, las técnicas clásicas no han producido, en algunos casos, los resultados deseados y se han mostrado efectivas sólo en las condiciones teóricas de diseño. Sin embargo, no sólo basta con desarrollar métodos que aprovechen las ventajas de las técnicas de Inteligencia Artificial, sino

que hay que aprovechar la sinergia entre éstas, los métodos computacionales avanzados y los avances tecnológicos en general. De este modo se pueden alcanzar diseños de sistemas de control y/o monitorización cualitativamente superiores y aplicarlos a la mejora de procesos productivos.

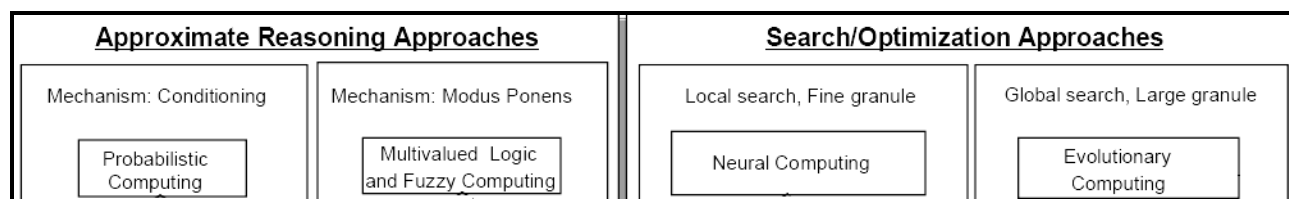
Dentro del amplio espectro de las técnicas de Inteligencia Artificial, una de las ramas con mayor desarrollo científico-técnico en los últimos años que se nutre, principalmente, de las sinergias entre las técnicas básicas de la IA es el “*Soft Computing*”. Esta abarca un amplio conjunto de métodos para el diseño de sistemas inteligentes, capaces de manejar de forma más provechosa información incierta, imprecisa y/o incompleta permitiendo, además, abordar problemas reales para obtener soluciones más robustas, manejables y de menor coste que las obtenidas mediante técnicas convencionales. El término *soft-computing* ha sido acuñado por Lofti Zadeh en una publicación en 1994 [4], definiéndola como (adaptación de la traducción realizada por Verdegay [5]):

"Básicamente, *soft-computing* no es un cuerpo homogéneo de conceptos y técnicas. Más bien es una mezcla de distintos métodos que de una forma u otra cooperan desde sus fundamentos. En este sentido, el principal objetivo del *soft-computing* es aprovechar la tolerancia que conllevan la imprecisión y la incertidumbre, para conseguir manejabilidad, robustez y soluciones de bajo costo. Los principales ingredientes del *soft-computing* son la lógica borrosa, la neuro-computación y el razonamiento probabilístico, incluyendo además como subconjunto a los algoritmos genéticos, las redes de creencia, los sistemas caóticos y algunas partes de la teoría de aprendizaje. En esa asociación de lógica borrosa, neuro-computación y razonamiento probabilístico, la lógica borrosa se ocupa principalmente de la imprecisión y el razonamiento aproximado, la neuro-computación del aprendizaje y el razonamiento probabilístico de la incertidumbre y la propagación de las creencias".

Zadeh define como premisas básicas del *soft-computing* que la imprecisión y la incertidumbre son características intrínsecas del mundo real, además que, la precisión y la certeza conlleva un coste. Expresa, además, como principio rector del *soft-computing*, explotar la tolerancia en la imprecisión, la incertidumbre y la verdad a medias, para alcanzar tratabilidad, robustez y soluciones de bajo coste [4].

Como se ha de extraer del concepto de Zadeh, la asociación de metodologías de cómputo que conforman el *soft-computing*, incluye como sus principales miembros [6] la lógica borrosa (LB), la neuro-computación (NC), la computación evolutiva (CE) y la computación probabilística (CP) (ver Figura 1.1). A partir de la simbiosis entre estas metodologías es posible diseñar sistemas inteligentes híbridos para ser aplicados a procesos físicos y de esta manera hacer frente a la imprecisión e incertidumbre inherente del mundo real.

En un trabajo publicado en 1999, Bonissone et ál. [7] ofrecen una amplia colección de sistemas inteligentes híbridos para realizar tareas de diagnóstico, estimación y control en aplicaciones industriales y comerciales, tales como dispositivos lógicos borrosos (DLB) ajustados mediante redes neuronales (NN, *Neural Networks*) y algoritmos de computación evolutiva (CE), redes neuronales ajustadas mediante CE o por DLB, además de algoritmos de CE controlados mediante dispositivos lógicos borrosos.



**Figura 1.1. Principales metodologías del *soft-computing* iniciales.**  
(Bonissone et ál., 1999 [7])

El *soft-computing* se mantiene en constante evolución y por tanto ha de inferirse que también sus componentes principales. El incremento de la potencia y el menor coste de los dispositivos de cómputo, además de los retos que plantea la complejidad de problemas reales, han promovido el incremento del empleo de algoritmos de tipo heurísticos, como valiosas herramientas capaces de proporcionar soluciones donde los algoritmos exactos no son capaces de encontrarlas [5, 8]. Estas técnicas heurísticas han demostrado su efectividad y se rigen por el principio común de la satisfacción como alternativa a la optimización, o dicho de otro modo según expresa el Prof. José L. Verdegay de la Universidad de Granada [5]: “[...] *antes que no poder dar la solución óptima a un problema, es mejor dar una solución que satisfaga al usuario en algún sentido que previamente habrá especificado*”.

Estos algoritmos heurísticos difieren en su fuente de inspiración, basándose en procesos naturales, sociales, físicos, entre otros, agrupándose por tanto en diferentes sub-clases de algoritmos. Varios de estos algoritmos son considerados metaheurísticos [9] debido a que su ejecución se produce mediante un proceso iterativo que guía una heurística subordinada, combinando de forma inteligente diferentes conceptos para explorar y explotar el espacio de búsqueda [10].

Algunos de estos algoritmos, por ejemplo, se agrupan dentro la denominada inteligencia de enjambre (SI, *Swarm Intelligence*) [11, 12], siendo los más populares el algoritmo de optimización de colonia de hormigas (ACO, *Ant Colony Optimization*) [13] y el algoritmo de optimización de enjambre de partículas (PSO, *Particle Swarm Optimization*) [14]. Estos algoritmos tienen como característica principal que describen el comportamiento de colectivos (ej.: colonias de hormigas, bancos de peces, bandadas de pájaros, etc.), ya sea la interacción entre los individuos que lo componen o con el entorno.

Otra gran sub-clase de algoritmos heurísticos constituye la que denominaremos algoritmos de simulación de procesos físicos (SPF), perteneciente a la gran clase de algoritmos de búsqueda de tipo Monte Carlo<sup>1</sup> [15]. Estos algoritmos gozan de una gran popularidad, constituyendo su principio de funcionamiento básico, la exploración del espacio de soluciones de un problema mediante alguna estrategia que transforme las soluciones de partida de forma iterativa. Estos algoritmos realizan un muestreo aleatorio iterativo para calcular sus resultados, simulando el comportamiento de diferentes sistemas físicos y matemáticos.

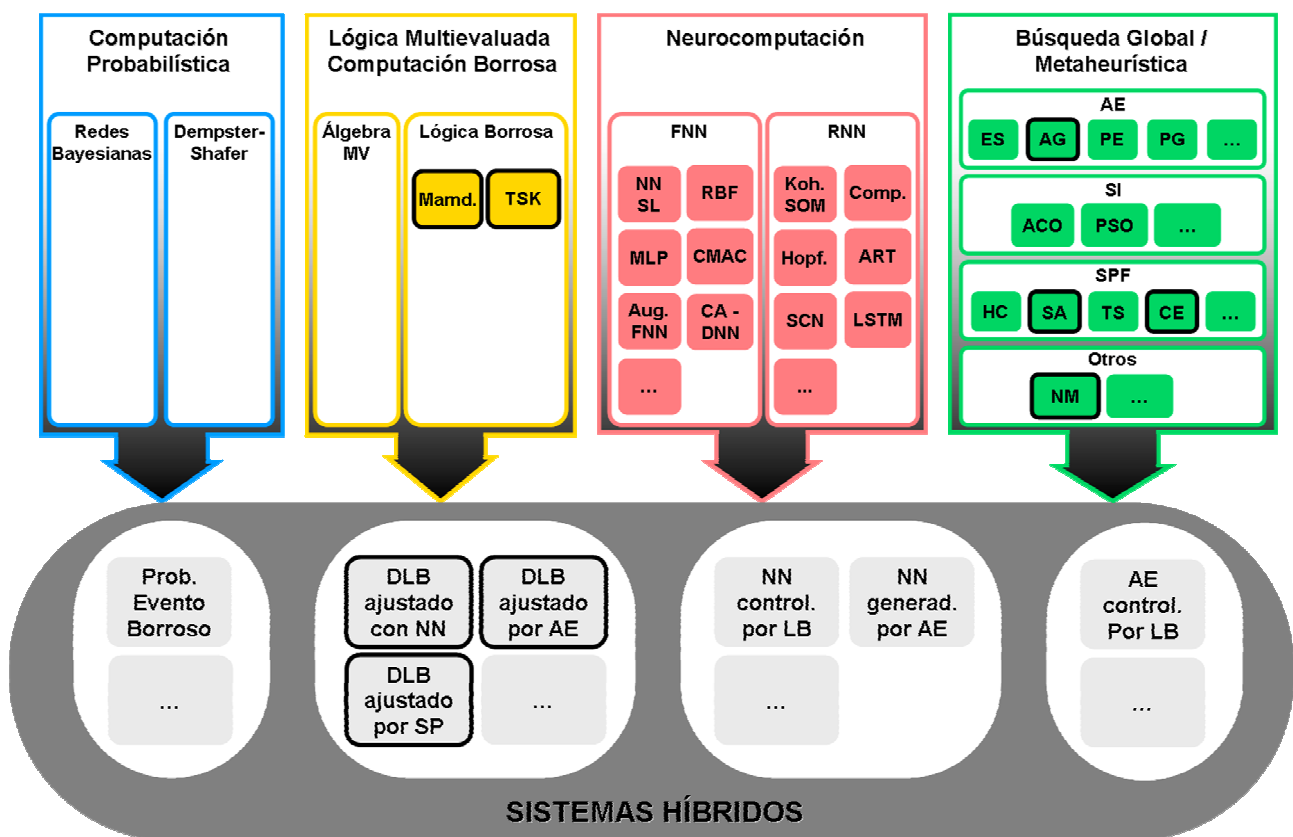


Figura 1.2. Componentes del *soft-computing* y sistemas inteligentes híbridos.

Entre los métodos de mayor popularidad que comparten este enfoque se encuentran el algoritmo de ascenso a colinas (HC, *Hill Climbing*) [16], temple simulado (SA, *Simulated Annealing*) [17, 18], búsqueda tabú (TS, *Tabu Search*) [19], entre otros. Otro de los algoritmos que vale la pena mencionar, aunque menos conocido debido a que su publicación

<sup>1</sup> La computación evolutiva y los algoritmos que la componen, como es el caso de los algoritmos evolutivos (AE), comparten el mismo principio que estos métodos y por tanto en la literatura son considerados también de tipo Monte Carlo (T. Weise, "Global Optimization Algorithms. Theory and Application," <http://www.it-weise.de>, 2009).



es más reciente y por ende su aplicación, es el algoritmo de entropía cruzada (CE, *Cross Entropy*) [20]. Este algoritmo fue desarrollado con el objetivo de simular eventos raros, con probabilidades muy pequeñas, pero con requerimientos de una buena precisión en su estimación.

Existe una amplia variedad de técnicas heurísticas y sub-clases de estas, que han sido aplicados y pueden emplearse de forma combinada con otros componentes del *soft-computing*, como son, además de las técnicas mencionadas con anterioridad, los algoritmos de búsqueda directa (ej.: algoritmo de optimización de Nelder-Mead (NM) [21]) y producto de la hibridación de métodos heurísticos (ej.: algoritmos meméticos), por solo mencionar algunos. La elección para la aplicación de una u otra técnica, dependerá de la eficiencia o desempeño que demuestre ante cada problema específico. Ofrecer una taxonomía completa de los algoritmos de búsqueda global, se escapa del ámbito de este trabajo y conllevaría una dispersión de los objetivos específicos que se persiguen.

La Figura 1.2 muestra un diagrama que pretende abarcar, tanto los miembros principales del *soft-computing*, varios de sus componentes básicos, así como sistemas híbridos que han sido propuestos en la literatura o que pueden surgir de la combinación de los componentes básicos. El diagrama parte de la clasificación inicial propuesta por Bonissone [7, 22], en la que se definen varios niveles. En este caso se ha situado en el primer nivel, los cuatro miembros principales actuales, la computación probabilística, la lógica multievaluada y computación borrosa, la neuro-computación y tal como proponen Verdegay, Yager y Bonissone [8], las técnicas metaheurísticas y algoritmos de búsqueda global, como cuarto componente, ya que abarca la computación evolutiva, tal y como había propuesto Zadeh inicialmente [6]. A continuación, para cada uno de los componentes básicos se relacionan algunas de sus técnicas más conocidas:

- Dentro de la computación probabilística se hace énfasis en las redes bayesianas y los sistemas Dempster-Shafer, para el desarrollo de modelos probabilísticos.
- Para el caso de la computación borrosa y multievaluada, en el diagrama se destacan por un lado los sistemas algebraicos de lógica multievaluada y los sistemas de inferencia borrosa, en concreto para este último, se especifican los dispositivos lógicos borrosos (DLB), empleados para aplicaciones de modelado y control de procesos, de tipo Mamdani [23] y de tipo Takagi-Sugeno-Kang (TSK) [24, 25].
- Dentro de la neuro-computación, algunas de sus estrategias básicas se han reagrupado de acuerdo a su topología de red, por un lado se relacionan las redes neuronales (NN, *Neural Networks*) artificiales anticipativas (FNN, *Feedforward NN*) y las redes neuronales recurrentes (RNN, *Recurrent NN*). Dentro de la categoría de redes FNN se detallan el perceptrón de capa simple (NN SL, *NN Single Layer*), el perceptrón multicapa (MLP, *Multiple Layer Perceptron*), las redes neuronales de base radial (RBF, *Radial Basis*

*Function*) [26], las redes conocidas como controlador del modelo aritmético del cerebelo (CMAC, *Cerebellar Model Arithmetic Controller*) [27] y con un desarrollo un poco más reciente, las redes neuronales aumentadas (*Augmented FNN*) [28], las redes neuronales dinámicas aditivas de tiempo continuo (CA-DNN, *Continuous-time Additive Dynamic NN*) [29, 30], entre otras. Relacionado con la categoría de redes RNN, estas han sido, por lo general, orientadas hacia el aprendizaje no-supervisado, como memorias asociativas y estructuras auto-organizadas. Algunas de las estructuras más conocidas son las redes competitivas, los mapas auto-organizados de Kohonen (Kohonen SOM, *Self-Organizing Map*) [31], las redes de Hopfield [32], los modelos de red basados en la teoría de la resonancia adaptativa (ART, *Adaptive Resonance Theory*) [33], las redes de almacenamiento a corto y largo plazo (LSTM, *Long Short Term Memory*) propuestas por Hochreiter y Schmidhuber [34] y las redes neuronales de conexión estocástica (SCN, *Stochastic Connection Neural networks*) [35], entre muchas otras propuestas.

- Relacionado con las técnicas de búsqueda global y metaheurísticas, las diferentes estrategias se han agrupado según las sub-clases de algoritmos mencionados con anterioridad, la inteligencia de enjambre (SI), los algoritmos de simulación de procesos físicos (SPF), por supuesto los algoritmos evolutivos (AE), que son considerados miembros del *soft-computing* desde su definición inicial y en este caso representados por los algoritmos genéticos (AG) [36], la programación genética (PG) [37], la programación evolutiva (PE) [38] y las estrategias de evolución (ES, *Evolution Strategies*) [39], resaltando además en el diagrama, el algoritmo de búsqueda directa de Nelder-Mead (NM).

En un segundo nivel del diagrama, se muestran algunos sistemas híbridos fruto de la combinación de las estrategias básicas. Por ejemplo, de la combinación de las técnicas probabilísticas con las estrategias borrosas, se han desarrollado los denominados modelos probabilísticos híbridos, fundamentalmente la probabilidad de eventos borrosos [40] y la veracidad de eventos borrosos [41], entre otros.

Por otro lado, la interacción de las estrategias de neuro-computación con las metodologías de la lógica borrosa y las técnicas de optimización, ha llevado a la elaboración de propuestas como los llamados sistemas neuronales híbridos, particularmente la generación formal de su estructura y los valores de los pesos en cada nodo, además del control borroso de los parámetros de las redes neuronales artificiales. Por ejemplo, en el caso de la aplicación de los algoritmos evolutivos, algunas propuestas han sido enfocadas hacia el entrenamiento optimizado de las redes, como alternativa a los métodos clásicos de entrenamiento (ej.: retro-propagación de errores) [42]. Además, las propuestas han tratado el tema de la evolución de la topología de la red, de elementos tales como sus capas ocultas, número de nodos por capas y enlaces entre ellos [43, 44], resaltando la idea de Maniezzo para evolucionar simultáneamente los pesos y la topología de la red [45]. En cambio, algunas aplicaciones para el control

borroso de las redes, han sido enfocadas a la monitorización del desempeño de las redes, al igual que a modificar los parámetros de control de las estrategias de aprendizaje. Ejemplos típicos han sido las propuestas de Jacobs [46] y Arabshahi et ál. [47], en las que se realiza un aprendizaje adaptativo mediante una ley de control borrosa, que varía la tasa de aprendizaje a partir del error obtenido durante el entrenamiento.

Más específicamente, para el caso de la idea de utilizar la lógica borrosa para mejorar el desempeño de los algoritmos evolutivos, las propuestas de hibridación han sido más recientes, desembocando principalmente en tres tipos de estrategias: los sistemas genéticos borrosos, los sistemas bioinspirados y el control borroso de los parámetros de algoritmos evolutivos. Las proposiciones realizadas van encaminadas principalmente a mejorar el procedimiento de ajuste de los parámetros que conforman cada una de las etapas de estos algoritmos, tales como el tamaño de la población, tasas de cruce, etc. Uno de los trabajos pioneros en este sentido, fue la proposición de Herrera y Lozano [48] para el ajuste de algoritmos genéticos, obteniendo una mejora en la eficiencia y la velocidad de convergencia.

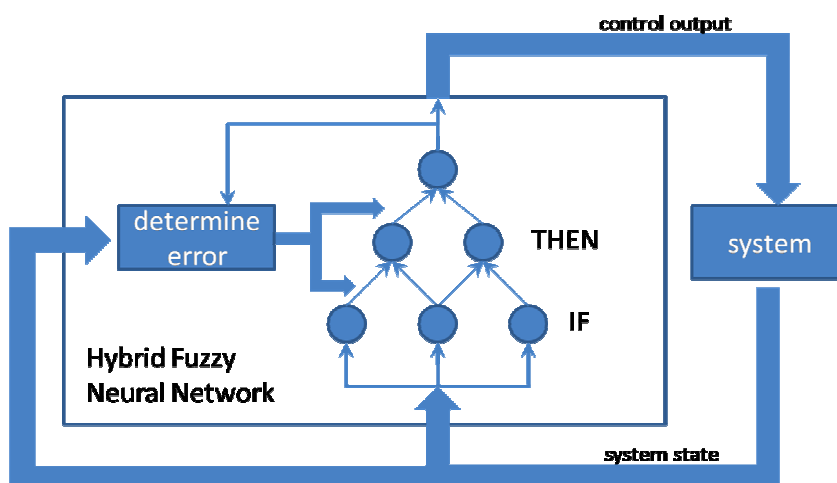


Figura 1.3. Sistema híbrido neuro-borroso para el control de procesos.  
(Kruse, 2008 [49])

Por otra parte, las iniciativas de hibridación entre estrategias borrosas con algunos de los componentes de la neuro-computación y las estrategias de optimización, al igual que la gran mayoría de los sistemas inteligentes híbridos, surgen de la necesidad de diseñar sistemas inteligentes que representen el conocimiento de forma eficiente computacionalmente, además de reducir costes de diseño. Por ejemplo, relacionado con los DLB de tipo Mamdani, en los cuales su base de conocimientos está formada por los factores de escala (FE) de sus variables de entrada y salida, las funciones de pertenencia (FP) también de cada una de las variables y la base de reglas (BR), se han desarrollado estrategias para el ajuste de estos componentes, ya sea mediante redes neuronales, algoritmos evolutivos y otras técnicas metaheurísticas. Para

este tipo de estrategias vinculadas con la lógica borrosa, se ha realizado un estudio más exhaustivo de las propuestas claves, debido a su mayor vínculo con los objetivos específicos de este trabajo.

En el caso de la hibridación con redes neuronales, gozan de gran popularidad los sistemas neuro-borrosos (ver Figura 1.3) que son, en esencia, sistemas de inferencia borrosa ajustados mediante redes neuronales o a través de sus estrategias de aprendizaje. Mediante esta estrategia la base de reglas se representa como una red neuronal, donde los parámetros de las funciones de pertenencia constituyen los pesos de las neuronas. Por tanto, al entrenar la red neuronal se ajusta la base de conocimientos del dispositivo lógico borroso. Algunos ejemplos de sistemas neuro-borrosos de tipo Mamdani disponibles en la literatura son el sistema NEFCON [50], desarrollado por Nauck en 1994 para tareas de control y el sistema NEFCLASS [51], propuesto por Nauck y Kruse para ejecutar labores de clasificación.

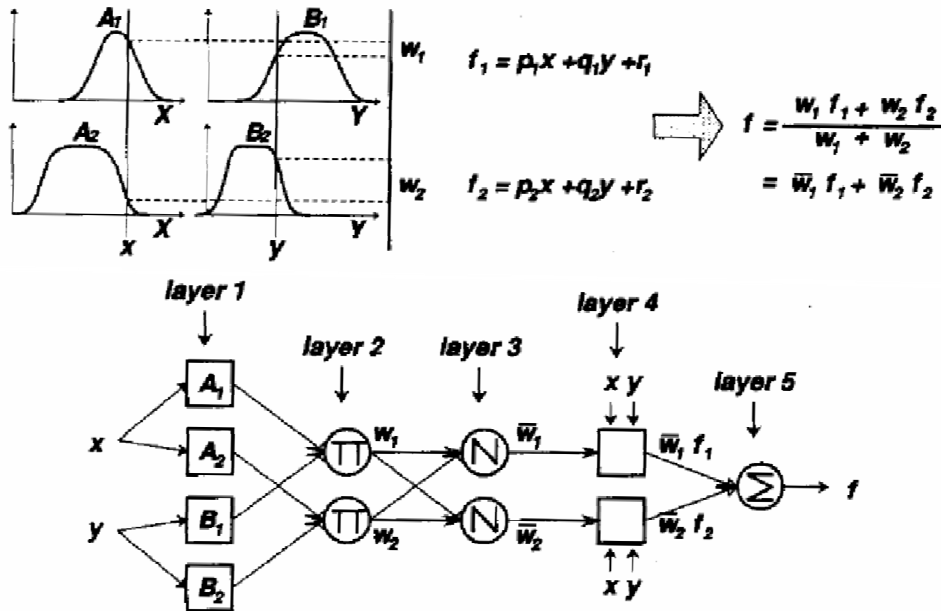


Figura 1.4. DLB de tipo TSK de primer orden y arquitectura ANFIS equivalente. (Jang, 1993 [52])

Por otra parte, en el caso del sistema de inferencia borrosa de tipo TSK, su estructura se ha equiparado con las redes neuronales de base radial [53, 54]. Este tipo de DLB emplea funciones polinómicas como salida en cada regla de su base de reglas, incrementando de esta forma, el potencial de representatividad del conocimiento. Basado en la equivalencia de ambas estructuras, en 1993 Jang publica el sistema de inferencia borrosa basado en redes neuronales adaptativas (ANFIS, *Adaptive Network-based Fuzzy Inference System*) [52] (ver Figura 1.4), donde la base de reglas determina la topología de la red, representando los nodos

en determinadas capas de la red, los parámetros de las funciones de pertenencia de las entradas y los coeficientes de los polinomios de salida de cada regla. En este trabajo Jang propone una estructura de red neuronal denominada redes adaptativas, las cuales no son más que redes neuronales artificiales multicapas con alimentación hacia delante, con la diferencia de que las funciones de activación pueden ser diferentes en cada nodo. Además, la red puede ser entrenada de forma híbrida, empleando el método del gradiente descendiente y el algoritmo de retropropagación de errores para ajustar los parámetros de los nodos de las capas ocultas, además del método de mínimos cuadrados para ajustar los parámetros de la capa de salida.

En cambio, de la combinación de la lógica borrosa con las técnicas de optimización, un ejemplo clásico de hibridación son los sistemas de inferencia borrosa generados y ajustados mediante algoritmos evolutivos. Para el caso del DLB en tareas de control, una estrategia fácil y práctica en su implementación es utilizar alguna función de coste que evalúe el comportamiento o desempeño del sistema de control a bucle o lazo cerrado. De esta manera a partir de los valores que alcanza la función de coste, empleando estrategias de optimización, puede ajustarse el DLB de forma óptima, ya sea por medio de las funciones de pertenencia o los factores de escala.

A principios de los años 90, se produce una avalancha de trabajos imposibles de ser recogidos todos en un capítulo de este trabajo, en los que se explora las potencialidades del empleo de los algoritmos genéticos para el ajuste de dispositivos lógicos borrosos, principalmente para el control de procesos. Este hecho fue recogido en una publicación en 1996 por Cordón et ál. [55], en el cual se relacionan más de 500 publicaciones de este tema. Algunas de las principales contribuciones constituyen por ejemplo, el trabajo de Karr de 1991 [56], uno de los precursores de esta estrategia de hibridación, proponiendo el ajuste de las funciones de pertenencia del DLB mediante algoritmos genéticos. Otro de los trabajos a resaltar es la propuesta de Lee y Takagi, publicada en 1992 [57], en la cual se ajustan las funciones de pertenencia y la base de reglas del DLB de tipo TSK, empleando cromosomas formados por la concatenación de los parámetros de las funciones de pertenencia del antecedente de cada regla y los coeficientes del polinomio del consecuente.

Otra de las propuestas claves, es el trabajo de Bonissone et ál. [58] inspirado en el procedimiento de Zheng para el ajuste manual del DLB [59]. Estos autores realizan el ajuste en tres etapas empleando algoritmos genéticos. Primeramente, a partir de una definición de las funciones de pertenencia de forma distribuida en todo el universo de discurso de cada una de las variables, realizan un ajuste de los factores de escala, tanto de entrada como de salida, causando un efecto “macroscópico” en la base de conocimientos del DLB, ya que los factores de escala definen los rangos de las variables dentro del universo de discurso y por tanto determinan el rango de las reglas, dentro de la base de reglas, que se activarán. Seguidamente, se efectúa un ajuste de las funciones de pertenencia, causando un efecto “mesoscópico” y

finalmente, si es necesario un mejor desempeño, se ajusta la base de reglas provocando un efecto “microscópico” en la base de conocimientos del dispositivo lógico borroso.

**Tabla 1.1. Aplicación del *soft-computing* al modelado, monitorización y control de procesos.**

<b>Técnica y aplicación</b>	<b>Autores</b>
Computación probabilística, Redes Bayesianas. Modelado y monitorización.	Correa et ál., 2008 [60]; Flynn y Todd, 2010 [61]; Elliott y Williams, 2006 [62]
Redes neuronales. Modelado y monitorización.	Alippi y Piuri, 1996 [63]; Coit et ál., 1998 [64]; Feng et ál., 2002 [65]
Redes neuronales. Modelado y control.	Barnes et ál., 1991 [66]; Hunt et ál., 1992 [67]; Jin et ál., 1994 [68]; Narendra et ál., 1990 [69]
Lógica borrosa. Modelado y control.	Babuska, 1998 [70]; Haber et ál., 2002 [71]
Sistemas híbridos neuro-borrosos. Modelado y control.	Santos et ál., 2006 [72]; Denai et ál., 2007 [73]; Sun et ál., 2005 [74]; Shari et ál., 1996 [75]; Liu et ál., 2004 [76]; Haber et ál., 2005 [77]
Algoritmos evolutivos, hibridación. Modelado y control.	Farag et ál., 1998 [78]; Carse et ál., 1996 [79]; Nearchou, 1999 [80], Martín et ál., 2009 [81]

En la Tabla 1.1 se resumen algunos trabajos disponibles en la literatura, en los cuales se aplican varias de las técnicas incluidas en el *soft-computing*, desde estrategias básicas hasta sistemas híbridos, para tareas de modelado, monitorización y control de procesos.

En este trabajo se hará uso de algunas de las ideas de hibridación propuestas en la literatura para ser consideradas en el control de procesos complejos. Partiendo de la lógica borrosa y principalmente del DLB como estrategia de control, se comprobará el desempeño de varios métodos metaheurísticos, pertenecientes a diferentes subclases de técnicas, para el diseño y ajuste óptimo de sistemas inteligentes para el control de procesos. En específico se explorarán las sinergias entre los algoritmos genéticos, el algoritmo de temple simulado, el algoritmo de Nelder-Mead y el algoritmo de entropía cruzada, el cual es una de las estrategia emergentes para el ajuste óptimo de sistemas de control. Se examinará además, la potencialidad de los sistemas híbridos neuro-borrosos, a través de los sistemas de inferencia borrosa ajustados mediante redes adaptativas, para el control de procesos. Las diferentes estrategias serán utilizadas para el control de procesos de mecanizado, de los cuales se realizará una breve introducción en la siguiente sección.

## 1.2 Operaciones de mecanizado como procesos electro-mecánicos complejos

Los procesos de mecanizado constituyen procesos de fabricación que abarcan un conjunto de operaciones de conformación de piezas o componentes, mediante la deformación y eliminación de material, ya sea por arranque de viruta o por abrasión. Estos procesos pueden realizarse empleando herramientas de uso manual, constituyendo un factor importante la fuerza y destreza para ejecutar las operaciones, que desempeñe el operador humano, pero por otra parte, y siendo los procesos a los que nos referiremos en este trabajo, también se llevan a cabo mediante máquinas herramientas, donde los esfuerzos necesarios para que se produzca la eliminación de material se produce por dispositivos electro-mecánicos.

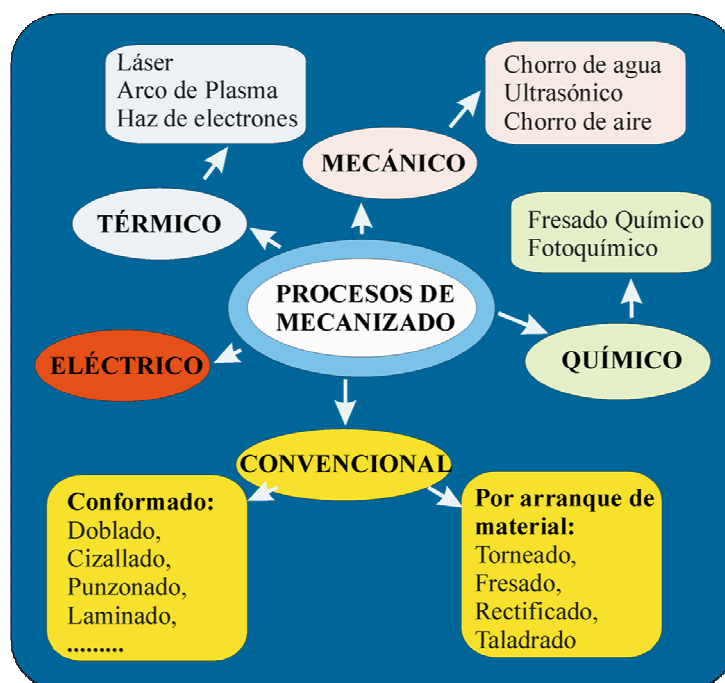
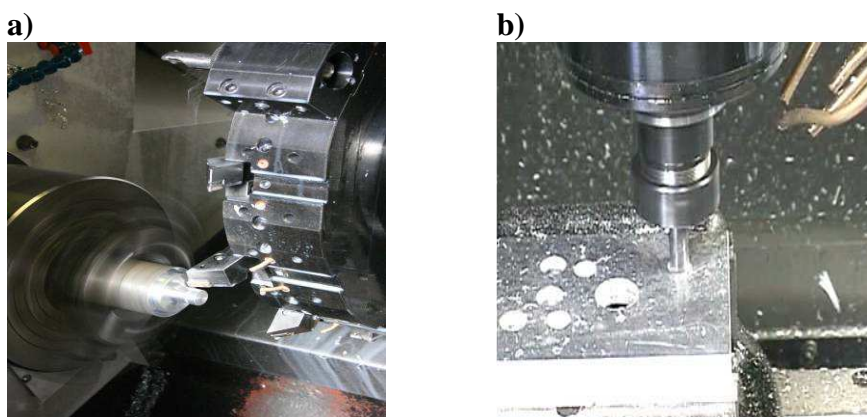


Figura 1.5. Clasificación de los procesos de mecanizado.

Las máquinas herramientas pueden operarse de forma manual (a través de servomecanismos) o de forma automatizada. Las primeras máquinas empleaban volantes para ajustar las posiciones y velocidades de sus ejes de movimiento y poseían complejos sistemas de engranajes y palancas para controlar las operaciones de mecanizado. A partir de la II Guerra Mundial y tras el incremento de la demanda en los procesos de fabricación, se desarrollaron los sistemas de control numérico. Las máquinas de control numérico iniciales, utilizaban tarjetas perforadas para las órdenes a su sistema de control. El desarrollo de las tecnologías de cómputo a partir de los años 60 y la incorporación de estos avances a las máquinas herramientas, provocó un incremento de la flexibilidad de los procesos de

mecanizado, surgiendo lo que conocemos hoy como máquinas de control numérico por computador o máquinas CNC. Estas máquinas son capaces de repetir secuencias de órdenes con precisión, pudiendo producir por tanto, piezas de mayor complejidad.

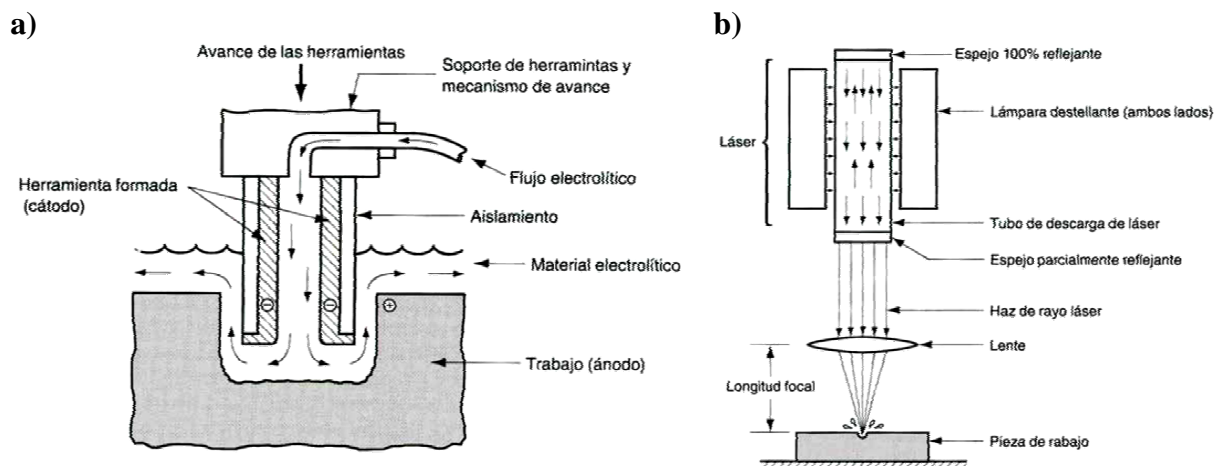
Los procesos de mecanizado se realizan sobre piezas, por lo general tratadas previamente, con el fin de mejorar sus propiedades mecánicas como la dureza, la resistencia, la elasticidad, etc. La Figura 1.5 muestra una clasificación de los procesos de mecanizado, agrupados principalmente por convencionales o no, y luego entre las operaciones tradicionales se hace diferenciación entre los procesos por arranque de viruta y procesos de conformado. Entre las operaciones que forman parte de los procesos de mecanizado por arranque de viruta (ver ejemplos de la Figura 1.6) se encuentran el taladrado, torneado, fresado, roscado, grabado, entre otras, las cuales son consideradas como operaciones de mecanizado convencionales. Constituyendo el taladrado, el torneado y el fresado, las operaciones con mayor peso en la industria.



**Figura 1.6. Ejemplos de operaciones de mecanizado convencional.**  
**a) Proceso de torneado y b) operación de taladrado.**

Los procesos de mecanizado tienen una gran importancia dentro de los sectores de fabricación. Estos procesos involucran maquinaria y materiales muy costosos, así como operadores expertos. Además, la fuerte competencia a nivel internacional en el sector del mecanizado obliga a desarrollar e implantar sistemas de monitorización y control que mejoren los procesos productivos en todas sus facetas, como pueden ser: incremento de las tasas de arranque del material, disminución de tiempos de parada, mejorar la calidad superficial de los componentes fabricados así como su calidad dimensional, conservar las herramientas de corte, además de la durabilidad de las máquinas, entre otras [82]. Sin embargo, la complejidad intrínseca de estos procesos hace que esta tarea no resulte nada sencilla. La fuerte interacción en las máquinas del movimiento relativo entre la pieza y la herramienta genera problemas como por ejemplo, vibraciones y el retemblado, la obtención de formas y de acabados imprecisos, deformaciones de orígenes mecánicos o térmicos, etc. [83].





**Figura 1.7. Ejemplos de operaciones de mecanizado no tradicional.**

**a) Mecanizado electro-químico y b) mecanizado con rayo láser (imágenes obtenidas de [84]).**

Por otra parte, el desarrollo de nuevas tecnologías ha traído consigo la implantación en los últimos años de nuevos procesos de mecanizado, en los cuales el arranque de material se produce de forma no convencional, empleando energía térmica, mecánica, química, eléctrica o la combinación entre ellas, denominándose operaciones de mecanizado no tradicionales (ver ejemplos de operaciones en la Figura 1.7). Estas operaciones suelen dividirse en cuatro categorías [85]:

1. Químicas: la eliminación de material se produce mediante una reacción química (ej., el mecanizado fotoquímico conocido también como fresado fotoquímico).
2. Eléctricas o electroquímicas: el material se elimina por disolución electrolítica. Ej., mecanizado electroquímico (ECM, *Electro-Chemical Machining*) y esmerilado electroquímico.
3. Térmicas: la operación se lleva a cabo concentrando altas temperaturas en zonas muy localizadas en las que funde y vaporiza el metal. Entre sus operaciones podemos citar el mecanizado por electro-erosión (EDM, *Electrical Discharge Machining*) donde la eliminación de material se produce mediante un arco eléctrico; el mecanizado mediante haz de electrones (EBM, *Electron-Beam Machining*) en el cual electrones a muy alta velocidad y energía inciden sobre la superficie de piezas, creando calor y evaporizando el material en la región de contacto; el mecanizado con rayo láser y con arco de plasma, entre otros.
4. Mecánicas: la eliminación de material se produce por erosión o corte con puntas múltiples. Ejemplo de operaciones, mecanizado ultrasónico, chorro de agua, mecanizado con chorro de agua abrasivo, entre otros.

El desarrollo de las ciencias de la computación y las tecnologías de los ordenadores, los dispositivos electro-mecánicos, los sistemas de medición avanzados, los nuevos materiales se ha producido de forma simultánea con el empuje, competencia y exigencias en general de la industria de fabricación. Estos avances han conducido a la evolución del mecanizado convencional al mecanizado de alto rendimiento (MAR) o también conocido como mecanizado de alta velocidad. Se trata de producir componentes de mayor valor añadido, demandando por tanto, una disminución de los costes y de los tiempos de producción, en paralelo con un aumento de la calidad de los componentes fabricados.

El concepto de mecanizado de alto rendimiento se refiere principalmente a operaciones por arranque de viruta en las cuales se han obtenido mejoras sustanciales en dos aspectos principales de los procesos [86]. Por un lado un incremento notable en la productividad, medida como tasa de arranque, es decir, cantidad de material eliminado por unidad de tiempo, sin que conlleve un desgaste excesivo de las herramientas y, por otro lado, la calidad que se obtiene de las piezas que se fabrican, reflejada en una mayor calidad superficial y dimensional.

Hoy en día los procesos de mecanizado de alto rendimiento imponen nuevos conceptos y estrategias de diseño mecánico en los elementos de la máquina herramienta, así como en el rediseño drástico de todos los esquemas relacionados con los sistemas de monitorización, control y supervisión de los procesos [87]. La complejidad de los procesos de mecanizado, dada por sus no-linealidades inherentes, su naturaleza multivariable, etc., se hace aún más patente en el nuevo concepto de mecanizado de alto rendimiento.

La constante evolución de estos procesos, demanda la necesidad de desarrollar e implementar estrategias basadas en la investigación y en la innovación que permitan, a nivel funcional, el aprovechamiento efectivo de la complementariedad y de las sinergias provenientes de diferentes áreas científicas y tecnológicas. Es por ello que los objetivos de este trabajo están encaminados al diseño de estrategias de control basadas en la hibridación de **técnicas de Inteligencia Artificial**, para la **optimización de procesos de mecanizado**. Esto será posible gracias al conocimiento generado sobre estos procesos mediante **representaciones computacionales** simples y eficientes, obtenidas, fundamentalmente, a partir de la aplicación de **técnicas de identificación experimental de sistemas dinámicos**.

### **1.3 Inteligencia Artificial para la optimización de los procesos de mecanizado**

Dada la complejidad de los procesos de mecanizado de alto rendimiento y por el hecho de que las técnicas de control que se aplican ofrecen prestaciones insuficientes, el desarrollo de sistemas inteligentes de fabricación y de estrategias de control inteligente para optimizar estos

procesos, constituyen una herramienta de muy gran utilidad para la continua mejora de estos procesos.

A lo largo de los últimos años, las aplicaciones de diferentes técnicas de Inteligencia Artificial (IA) a las operaciones de mecanizado y de fabricación en general, han estado enfocadas principalmente a su optimización mediante la manipulación de los parámetros que determinan las operaciones de corte (parámetros de corte), el desarrollo de sistemas de monitorización, supervisión y control y sistemas de diagnóstico y detección de fallos, produciendo resultados sorprendentes. De las estrategias desarrolladas, las que se centran en los parámetros de corte han sido las más estudiadas y las que mejores resultados han aportado.

Según la literatura a la que se ha tenido acceso, tanto las redes neuronales artificiales (RNA) como los sistemas de inferencia borrosa han sido, de las técnicas de Inteligencia Artificial, las que con mayor frecuencia se han empleado para la optimización de los procesos de mecanizado. Debido a su capacidad para la aproximación de funciones, en especial de relaciones no-lineales, las redes neuronales artificiales han resultado ser una técnica muy atractiva para el modelado de los procesos de corte [88, 89].

En un trabajo publicado en 1998, Coit et ál. [64] consideraron los aspectos prácticos de la construcción y validación de modelos neuronales, mientras que Viharos et ál. [90] mostraron la aplicación de las redes neuronales artificiales al control y monitorización de procesos de mecanizado. Diversas aplicaciones de modelado de procesos de mecanizado a través de redes neuronales artificiales basadas en relaciones entradas-salidas han sido reportadas en la literatura. Ejemplo de ello constituyen las aplicaciones satisfactorias del perceptrón multicapa (MLP) para modelar el lijado de super-aleaciones propuesta por Sathyanarayanan et ál. [91], el modelo desarrollado por Jain et ál. [92] para la predicción de la tasa de arranque de material y el acabado superficial de un mecanizado abrasivo, además del modelo propuesto por Feng et ál. [65] del tallado de cilindros para motores. Además, en un trabajo desarrollado por Grzesik y Brol [93], estos autores mostraron la utilidad del modelado y el control neuronal, hibridado con otras técnicas de análisis (ej., estadísticas), de ciertas características del acabado superficial en procesos de mecanizado de múltiples etapas. Algunos de los inconvenientes de las redes neuronales artificiales son que los parámetros del modelado no pueden ser interpretados, que son dependientes de grandes conjuntos de datos de entrenamiento, que ciertas configuraciones de las redes producen sobre-entrenamiento y que, en ocasiones, el entrenamiento culmina sin que se hayan alcanzado soluciones óptimas.

Por otro lado, la lógica borrosa también ha jugado un papel muy importante en el modelado de procesos de mecanizado. El modelado basado en la teoría de conjuntos borrosos es usualmente utilizado cuando existe conocimiento subjetivo del proceso u opiniones de operadores expertos, desempeñando un papel fundamental en la definición de funciones objetivo y variables de decisión. Shin y Vishnupad [94] observaron que la hibridación de los

sistemas de inferencia borrosa con las técnicas de neuro-computación para el modelado de procesos, podría ser una manera efectiva de controlar procesos de mecanizado como la operación de pulido. Para esta misma operación y empleando esta misma técnica de hibridación, Chen y Kumara publicaron en 1998 [95] un sistema inteligente híbrido neuro-borroso para controlar el proceso. Esta idea de integración de ambas técnicas también fue resaltada por Kuo y Cohen para el control efectivo de procesos de fabricación [96]. Cabe destacar también en este campo el trabajo de Kamatala et ál., en el cual se desarrolla un sistema basado en lógica borrosa para predecir la rugosidad superficial en procesos de torneado [97] y la propuesta de Hashmi et ál. de selección de los parámetros de corte en operaciones de fresado mediante sistemas de inferencia borrosa [98].

**Tabla 1.2. Técnicas de IA en la optimización de procesos de mecanizado.**

<b>Modelado, Monitorización y Supervisión</b>	<b>Relacionado con</b>	<b>Autores</b>
Modelo basado en algoritmos genéticos para la predicción de la fuerza de corte en operaciones de <b>fresado</b> .	Herramienta	Milferner et ál., 2005 [99]
Modelos teóricos y experimentales del proceso de <b>rectificado</b> basados en la LB y RNA.	Calidad Superficial	Maksoud y Atia, 2004 [100]
Estimación de la condición de la herramienta de corte utilizando SOM en <b>torneado</b> .	Herramienta	Rao y Srikant, 2004 [101]
Monitorización del proceso de <b>torneado</b> basada en: filtrado de características redundantes por medio de agrupamiento borroso y fusión sensorial empleando RNA.	Herramienta	Fu y Hope, 2004 [102]
Modelo borroso TSK para la estimación de las condiciones de <b>rectificado</b> en las que se produce quemaduras.	Herramienta	Ali y Zhang, 2004 [103]
Sistema borroso en el <b>fresado</b> para la supresión de vibraciones.	Vibraciones	Liang et ál., 2004 [104]
RNA anticipativa en operaciones de <b>fresado</b> .	Herramienta	Haber y Alique, 2003 [105]

Por otra parte, en 1998 W.L.R. Ip publicó una estrategia de control basada en lógica borrosa para la regulación de la velocidad de avance en el fresado de un acero templado, obteniendo una mayor eficiencia de corte y un mayor aprovechamiento de la vida útil de la herramienta [106]. Lee et ál. usaron la teoría de conjuntos borrosos para obtener un modelo no-lineal del proceso de torneado, mostrando de este modo una herramienta más efectiva que las técnicas de modelado matemático para el caso en el que exista cierta “vaguedad” en la variables de control del proceso [107]. Al-Wedyan et ál., ya en el año 2001, usaron técnicas de modelado borroso para operaciones de fresado [108]. Para una mayor ilustración de las

aplicaciones de las técnicas de IA para la optimización de procesos de mecanizado, en la Tabla 1.2 se resumen otras aportaciones recogidas en la literatura.

Con un mayor enfoque hacia el tipo de operación de mecanizado, al cual se ha aplicado la técnica y clasificados los procesos en operaciones tradicionales y no convencionales, en la Tabla 1.3 también se resumen varias de las propuestas y aportaciones de los últimos años.

**Tabla 1.3. Sistemas inteligentes en la automatización de procesos de mecanizado.**

	<b>Sistemas Inteligentes en la Automatización</b>	<b>Operación</b>	<b>Autores</b>
<b>Mecanizado no convencional</b>	Optimización de la trayectoria utilizando colonias de hormigas.	Punzonado y corte con láser	Wang y Xie, 2005 [109]
	Control borroso de la conductividad eléctrica de la disolución (simulación).	Mecanizado electroquímico	Skrabalak et ál., 2004 [110]
	Control borroso de la distancia entre los electrodos	Electro-erosión	Behrens y Ginzl, 2003 [111]
	MLP para el modelado del proceso.	Mecanizado químico	Shi, Schillings y Boyd, 2004 [112]
	MLP para el modelado del proceso.	Micro-mecanizado con láser	Yousef et ál., 2003 [113]
	ANFIS para predecir la rugosidad superficial	Pulido mecánico-químico	Lin y Liu, 2003 [114]
<b>Mecanizado tradicional</b>	Control neuro-borroso del esfuerzo de corte basado en el control por modelo interno.	Fresado de alta velocidad	Haber et ál., 2005 [77]
	Control neuro-borroso de la deflexión de la pieza y compensación de las deformaciones.	Rectificado	Ding et ál., 2004 [115]
	RNA para generar programas piezas en el taladrado.	Taladrado	Balic, 2004 [116]
	Control neuro-borroso de la fuerza de corte utilizando el aporte basado en caja gris.	Torneado	Lian et ál., 2005 [117]

A pesar de todas las aplicaciones desarrolladas, el empleo aislado de sistemas borrosos diseñados a partir del conocimiento de operadores expertos, no resulta óptimo para el modelado de los cambios dinámicos subyacentes en cualquier proceso de mecanizado. El aprovechamiento de las sinergias con otras técnicas básicas del *soft-computing*, y por tanto de la IA, permitirán, como ya se ha demostrado por algunas de las aportaciones recogidas en la literatura, el diseño de sistemas inteligentes que aprovechen de forma óptima el conocimiento de estos expertos, que en ocasiones puede carecer de precisión y completitud, y que permitan, además, enfrentar los problemas de complejidad de los procesos de mecanizado mediante soluciones robustas y de menor coste.

En este sentido, la actividad investigadora, que ha servido como base para el desarrollo de esta Tesis Doctoral, se ha enfocado al desarrollo de estrategias y algoritmos basados en la

conjunción de paradigmas de las Ciencias de la Computación y la Inteligencia Artificial y de la Automática. Esta actividad dio sus primeros pasos en el marco de los proyectos de investigación COREMAV [118], COREFAV [119] y NANOCUT-INT [120], en los cuales se han diseñado e implementado sistemas de monitorización y control en arquitecturas distribuidas, para la optimización de procesos de fabricación complejos. Los sistemas se han basado, fundamentalmente, en el empleo de técnicas de Inteligencia Artificial y se han aplicados a procesos tales como el fresado de alta velocidad, el taladrado de alto rendimiento y procesos de corte en la micro-escala, tal como el micro-torneado.

Durante el desarrollo de las arquitecturas de monitorización y control se han aprovechado varias sinergias. Por un lado, la sinergia entre la utilización del conocimiento obtenido por la información sensorial y el procedente de operadores y técnicos expertos. Por otro lado, las ventajas de la conjunción entre técnicas clásicas de control y de Inteligencia Artificial, además de la asociación entre algoritmos computacionales avanzados en sistemas intermediarios y tecnologías de comunicaciones. Esto ha dado lugar al desarrollo de una arquitectura basada en un sistema intermediario y en un sistema operativo de tiempo real, para incorporar sistemas de control y monitorización distribuidos. Se ha obtenido, además, un método de diseño de sistemas de control de inferencia borrosa óptimos, basados en el paradigma del control por modelo interno, para ser aplicados al control en sistemas distribuidos, a lazo cerrado, de procesos de fabricación.

Gracias a los conocimientos científicos y técnicos adquiridos en los proyectos citados con anterioridad, en los que se ha participado de forma activa, se evidenció la necesidad de desarrollar nuevas capacidades en los sistemas inteligentes artificiales para el control de procesos complejos de fabricación. Planteándose como nuevo reto científico-técnico, el diseño e implementación de sistemas de control con capacidades cognitivas artificiales, es decir, capacidades de auto-optimización, auto-adaptación, auto-configuración y auto-aprendizaje eficientes. Es por ello que la actividad investigadora que sustenta el desarrollo de esta Tesis Doctoral, se ha dirigido al desarrollo de métodos computacionalmente eficientes para el ajuste óptimo de sistemas de control inteligente, capaces de ser integrados en sistemas intermediarios con especificaciones de tiempo real. Estos métodos constituyen la base para el desarrollo de los sistemas de control inteligente con capacidades cognitivas.

En la actualidad la actividad investigadora se realiza bajo el paraguas de los proyectos de investigación COGNETCON [2] y MONINTDRILL [121], siendo el objetivo principal de ambos el desarrollo de un método para el diseño de un sistema de control cognitivo en red, basado en modelos internos, para procesos de mecanizado de alto rendimiento. Mediante estos proyectos se pretende evolucionar los controladores inteligentes a partir de la imitación de la evolución de la mente biológica, incorporando estrategias de competición, aspecto clave en los sistemas cognitivos.

## 1.4 Conclusiones

En este capítulo se ha realizado una revisión del estado de la ciencia y la técnica en los paradigmas elegidos dentro del amplio universo que ponen a nuestra disposición la Teoría General de Sistemas, la Teoría de Control, las técnicas de Inteligencia Artificial y las Ciencias de la Computación. Se ha prestado especial interés en los paradigmas con una fundamentación teórica sólida en el campo de la Inteligencia Artificial (IA), con un enfoque desde la perspectiva de la Automática y la Teoría de Sistemas. El estudio se ha centrado en estrategias susceptibles de ser utilizadas en el modelado, monitorización y control de procesos físicos. Paradigmas que aprovechan tanto las ventajas de las técnicas básicas de Inteligencia Artificial, como las sinergias entre estas. Por este motivo, la revisión se ha enfocado hacia las técnicas del *soft-computing*, analizando su evolución desde sus inicios hasta las tendencias actuales.

Debido a que las estrategias serán aplicadas al control de procesos electro-mecánicos complejos, como es el caso de los procesos de mecanizado, se ha realizado además, una breve revisión del estado de la técnica de estos procesos. Presentándose de forma breve como ha sido su evolución hasta nuestros días y las tendencias actuales en la aplicación de técnicas de IA, desde la perspectiva del control, para hacer frente a la complejidad de estos procesos y así mejorar su eficiencia.

En el siguiente capítulo se describe la aplicación de métodos de investigación tanto teóricos como experimentales para el modelado, mediante algoritmos computacionales eficientes, de algunos de estos procesos, como es el caso del fresado de alta velocidad y el taladrado de alto rendimiento.





---

## **CAPÍTULO 2**

# **UN ENFOQUE COMPUTACIONAL PARA EL MODELADO DE PROCESOS DE FABRICACIÓN**

---

Uno de los aspectos claves, al cual la comunidad científica continúa prestándole atención en la actualidad, es la obtención de modelos que permitan caracterizar los sistemas físicos presentes en una gran variedad de procesos y con ello poder simular su comportamiento [122-126].

Los sistemas de supervisión y control de procesos complejos, desempeñan un papel cada vez más relevante para alcanzar un cambio cualitativo en los procesos. Para ello, es necesario disponer de métodos de cómputo eficientes para la predicción ó estimación con cierto grado de precisión, el posible comportamiento temporal de sus variables de salida, a partir de sus variables de entrada y sus parámetros. La monitorización del comportamiento dinámico de estas variables permitiría detectar fallos en el proceso, desgaste de los elementos actuadores, predecir la presencia de perturbaciones no deseadas y aportar información acerca de la eficiencia y calidad de los procesos. Para ello resulta clave la obtención de un modelo que permita la simulación de procesos físicos y la predicción del comportamiento sus variables más significativas.

Una de las herramientas más ampliamente utilizadas dentro del campo de las ingenierías, para el análisis y modelado de sistemas y procesos físicos, es Matlab<sup>®</sup>. Esta herramienta es un lenguaje de alto rendimiento para la computación técnica, que integra el cómputo, la visualización y la programación en un ambiente fácil de uso donde los problemas y las soluciones se expresan en una notación matemática familiar. En el entorno se incluyen facilidades para el análisis numérico, cálculo matricial, procesamiento de señales, visualización gráfica y desarrollo de aplicaciones, incluyendo el diseño de interfaz gráfica de

usuario. Simulink es un paquete de programa de Matlab y se utiliza para modelar, simular y analizar sistemas dinámicos. También tiene soporte para sistemas lineales y no lineales, ya sean discretos o continuos en el tiempo.

Su elección para la implementación de modelos de procesos de alta complejidad, como son los procesos de fabricación, está motivada por la sencillez de la programación, la posibilidad de realizar simulaciones y aplicaciones en tiempo real, la portabilidad de los programas desarrollados y la gran potencia de cálculo que posee este entorno. Simulink es además un entorno de desarrollo muy práctico en ingeniería con una biblioteca de métodos numéricos y la posibilidad de modificar un gran número de parámetros de una manera amigable y sencilla. De ahí su amplia utilización para el modelado y optimización de problemas de nuestro entorno real.

Este capítulo presenta el diseño e implementación en Matlab/Simulink de modelos de procesos de fabricación complejos. Los modelos implementados son obtenidos a partir de las ecuaciones físico-matemáticas de los procesos, así como mediante el empleo de técnicas de identificación experimental. El capítulo está organizado en cinco apartados. La sección 2.1 introduce la técnica de identificación de sistemas dinámicos empleada y se describen varias de las herramientas y algoritmos utilizados durante los procedimientos de identificación, presentando, además, una plataforma experimental para estudiar procesos de fabricación. La sección 2.2 presenta el diseño e implementación de un modelo computacional del comportamiento de las variables más representativas del proceso de fresado de alta velocidad, a partir de su modelo físico. En las secciones 2.3 y 2.4 se describe el diseño e implementación, de modelos dinámicos de procesos de fabricación complejos, obtenidos mediante técnicas de identificación experimental, mostrando en cada una de las secciones, todo el procedimiento de identificación empleado. Finalmente, se plantean las conclusiones sobre los resultados obtenidos.

## **2.1 Identificación de sistemas dinámicos**

El modelado de sistemas dinámicos constituye una práctica muy común en los sistemas de monitorización y control de procesos electromecánicos complejos. Una de las técnicas más ampliamente utilizadas para el modelado matemático de procesos o sistemas, es la identificación experimental, que representa en sí misma todo un campo para el modelado matemático de sistemas dinámicos, a partir de la observación y medición de sus variables de entrada y de salida [127, 128].

El procedimiento de identificación de sistemas (ver Figura 2.1), según Ljung [127], está formado, principalmente, por los siguientes pasos básicos: experimentos para la identificación, selección de la estructura del modelo, estimación de los parámetros del modelo y la validación del modelo.

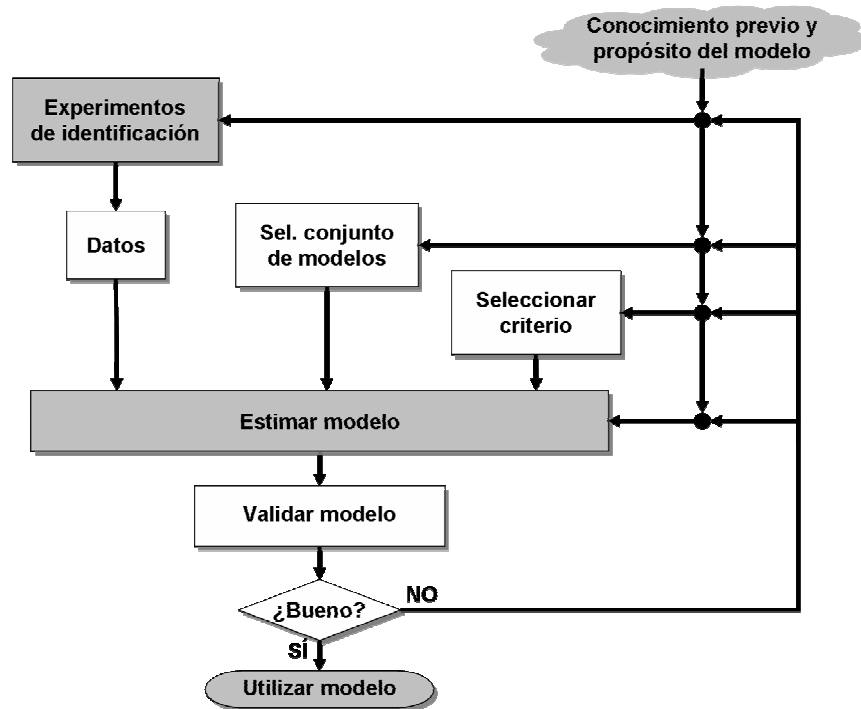


Figura 2.1. Procedimiento de identificación.  
(Imagen reformada a partir de Zhu, 2001 [128])

### *Identificación experimental de procesos simple entrada-simple salida*

A partir del comportamiento temporal de las variables en un proceso lineal e invariante en el tiempo, simple entrada – simple salida (SISO), es posible identificar características dinámicas de este, como por ejemplo, si se comporta como un sistema de primer orden, de segundo, etc. Teniendo una idea del orden del sistema, es posible aplicar varios métodos de identificación para obtener el modelo que más se ajuste al proceso. Sin embargo, un método para la identificación experimental más ampliamente utilizado, debido a su sencillez y robustez, constituye el análisis de respuesta ante una entrada a escalón del proceso a lazo abierto, es decir, sin realimentación de su salida. El método consiste en aplicar un cambio de tipo escalón en la variable de entrada del proceso (variable manipulada) y obtener el comportamiento de la variable de salida (variable controlada), midiendo y registrando las señales que caractericen a estas variables para su posterior análisis.



Figura 2.2. Respuesta a escalón de un proceso no realimentado.

Un modelo de un proceso SISO de tiempo continuo, lineal e invariante en el tiempo, puede describirse a través de una función de transferencia mediante el operador de Laplace  $s$ . La función de transferencia se define a partir del conocimiento de los parámetros del sistema como sus ceros, sus polos y el posible retraso de tiempo  $\tau$ , tal como se muestra en la siguiente ecuación:

$$G(s) = \frac{y(s)}{u(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_m}{s^n + a_1 s^{n-1} + \dots + a_n} \cdot e^{-s\tau} \quad (2.1)$$

Esta función de transferencia puede expresarse como la siguiente ecuación diferencial generalizada:

$$y(t) = \frac{1}{a_n} \left[ \left( b_0 \frac{d^m u(t-\tau)}{dt^m} + b_1 \frac{d^{m-1} u(t-\tau)}{dt^{m-1}} + \dots + b_m u(t-\tau) \right) - \left( a_{n-1} \frac{dy(t)}{dt} + \dots + a_1 \frac{d^{n-1} y(t)}{dt^{n-1}} + \frac{d^n y(t)}{dt^n} \right) \right] \quad (2.2)$$

Al expresar de forma discreta la ecuación diferencial anterior empleando un tiempo de muestreo  $T$ , se obtiene la ecuación en diferencias correspondiente y esta a su vez se puede expresar según la siguiente relación lineal, donde  $t = kT$ :

$$y(k) = x_1(k)\theta_1 + x_2(k)\theta_1 + \dots + x_p(k)\theta_p \quad (2.3)$$

Evalutando la ecuación anterior para  $P$  muestras en el intervalo de tiempo  $t = T, 2T, \dots, PT$ , se obtiene un sistema de ecuaciones lineales, el cual se representa de forma matricial en la siguiente ecuación:

$$\mathbf{Y} = \mathbf{X} \cdot \boldsymbol{\theta}, \text{ donde } \mathbf{Y} = \begin{bmatrix} y(1) \\ \vdots \\ y(P) \end{bmatrix}, \mathbf{X} = \begin{bmatrix} x_1(1) & \dots & x_p(1) \\ \vdots & \ddots & \vdots \\ x_1(P) & \dots & x_p(P) \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_p \end{bmatrix} \quad (2.4)$$

Los parámetros del modelo  $\hat{\boldsymbol{\theta}}$  pueden estimarse a partir de la ecuación (2.5). Una solución directa para el sistema de ecuaciones usualmente no es posible, dado que típicamente el número de ecuaciones, determinado por el total de muestras  $P$ , es mayor que el total de parámetros a determinar  $p$  (sistema sobredeterminado).

$$\hat{\boldsymbol{\theta}} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{Y} \quad (2.5)$$

A partir de la aplicación del algoritmo de optimización de Newton-Gauss estándar para la minimización del error residual  $e(t)$ , obtenido mediante la ecuación (2.6), es posible estimar los parámetros del modelo. Si la convergencia no es suficiente, se aplicaría el algoritmo de optimización de Levenberg-Marquardt para la minimización del error. La combinación de estos dos algoritmos de optimización es utilizada satisfactoriamente en una amplia variedad de aplicaciones y también ha sido extendido a la identificación de sistemas de control no lineales [129]. El procedimiento de identificación resumido anteriormente, ha sido descrito en detalle por Zhu [128].

$$e(t) = y(t) - \hat{y}(t) \quad (2.6)$$

### 2.1.1 Pre-procesamiento de los datos para la identificación experimental

Un paso no poco importante dentro del procedimiento de identificación, es el pre-procesamiento de los datos medidos. La precisión del modelo a ajustar depende en gran medida de la calidad de los datos empleados para la identificación, entendiéndose por calidad de los datos, cuestiones como el nivel de ruido, empleo de los rangos de frecuencias adecuadas que aportan la información necesaria para el modelo, cambios inesperados del nivel de las señales o de su tendencia provocado por cualquier perturbación durante la adquisición de los datos, etc. Además de los fenómenos que pueden aparecer en las señales eléctricas, introducidos por el propio sistema de medición y adquisición y que no guardan relación con el comportamiento físico del proceso bajo estudio, como por ejemplo, las *derivadas* de los sensores y amplificadores, sub-muestreo o solapamiento (*aliasing*) en las señales debido al procedimiento de muestreo, ruido introducido en la señal debido a la respuesta dinámica de los sensores amplificadores, entre otros.

Entre las técnicas más ampliamente empleadas durante el pre-procesamiento, se encuentra el filtrado de los datos, el cual permite eliminar de las señales los rangos de frecuencias en los que no es de interés modelar el comportamiento dinámico de los procesos o no aportan información significativa. Al aplicar un algoritmo de filtrado a las señales medidas, se debe tener en cuenta que las respuestas dinámicas de los filtros causales, afectan tanto a la amplitud de la señal a filtrar como a sus características de fase, por lo que el algoritmo de filtrado introduce retrasos de tiempo en la señal filtrada y por supuesto, ha de influir en el modelo final que se obtenga.

El filtrado con fase cero, no introduce cambio de fase en la señal que se procesa, pero este tipo de filtro no cumple el principio de causalidad, por lo que no es posible su implementación en tiempo real pero, por otra parte, puede utilizarse fuera de línea durante el pre-procesamiento de los datos. La Figura 2.3 muestra un diagrama del algoritmo de filtrado [130], en el cual a una secuencia de datos que se desea filtrar  $x[k]$ , se le aplica un filtro causal

con una característica de fase arbitraria y respuesta a impulso  $h[k]$ , obteniéndose la secuencia de datos  $g[k]$ . Seguidamente la nueva secuencia es invertida ( $g[-k]$ ), aplicándose nuevamente el mismo filtro y dando lugar a la secuencia de datos  $r[k]$ . Por último, esta secuencia se invierte y se obtiene  $s[k]$ , que constituye la secuencia filtrada pero sin cambio de fase con respecto a  $x[k]$ .

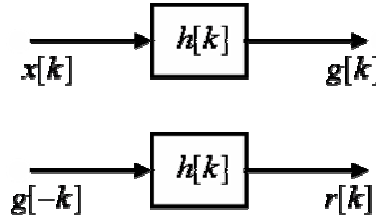


Figura 2.3. Algoritmo de filtrado con fase cero.

En las ecuaciones (2.7) y (2.8) se describe el algoritmo de filtrado, donde  $H(e^{j\omega})$  es la transformada de Fourier de la respuesta a impulso del filtro  $h[k]$ ,  $h_1[k]$  representa la respuesta total a impulso del filtro con fase cero y  $H_1(e^{j\omega})$  su transformada de Fourier.

$$\begin{aligned} g[k] &= x[k] * h[k] \\ r[k] &= g[-k] * h[k] \end{aligned} \quad ; \quad s[k] = r[-k]; \quad s[k] = x[k] * h_1[k] \quad (2.7)$$

$$\begin{aligned} h_1[k] &= h[k] * h[-k] \\ H_1(e^{j\omega}) &= H(e^{j\omega}) * H(e^{j\omega}) = |H(e^{j\omega})|^2 \end{aligned} \quad (2.8)$$

Por otra parte, un algoritmo de filtrado muy sencillo pero de gran utilidad para el procesamiento de señales periódicas, es el filtro de respuesta a impulso finita (FIR) de media móvil. El principio de este filtro reside en hallar la media de las muestras de una secuencia dada, utilizando un total de muestras igual a la ventana de datos  $ws$ . La respuesta a impulso del filtro se describe en la ecuación (2.9).

$$h[k] = \frac{1}{ws} \left[ 1 + k^{-1} + k^{-2} + \dots + k^{-(ws-1)} \right] \quad (2.9)$$

La aplicación de este filtro a una señal periódica permitiría obtener el valor medio de la señal o su componente de directa o de muy bajas frecuencias. Entre los procesos que se encuentran caracterizados por este tipo de señal, se encuentran los procesos de fabricación, donde señales de gran importancia como las fuerzas de corte, poseen componentes periódicas debido a la influencia de los accionamientos presentes en estos procesos.

### 2.1.2 Plataforma experimental para la identificación de procesos de fabricación

Con vistas a implementar estrategias de identificación en procesos y realizar una adecuada validación de modelos, se ha utilizado una plataforma experimental instalada en laboratorio para estudiar procesos de fabricación. La plataforma está constituida principalmente por un centro de mecanizado de alta velocidad KONDIA HS1000 equipado con CNC abierto Siemens 840D (ver Figura 2.4). Algunas de las características técnicas de este centro de mecanizado son las siguientes: 26 kW de potencia, velocidad de giro máxima del cabezal igual a 24000 rpm, máxima velocidad de avance de la máquina igual 24 m/min y un par máximo disponible de 28 Nm @6000 rpm. Para medir fuerzas durante las operaciones de mecanizado, se utiliza una plataforma dinamométrica Kistler 9257, instalada en la bancada de la máquina. Las características técnicas más importantes de esta plataforma dinamométrica son: un rango de medición [-5, 5] kN en cada eje, frecuencia natural > 4kHz, linealidad  $\leq 2\%$  a escala completa y una sensibilidad de -7.5 pC/N. Se utilizan además 3 amplificadores de carga 5011, cada uno para medir fuerza en los tres ejes cartesianos,  $F_x$ ,  $F_y$  y  $F_z$ . La adquisición de estas señales se realiza por medio de una tarjeta de adquisición de datos DAQBOARD-2005.

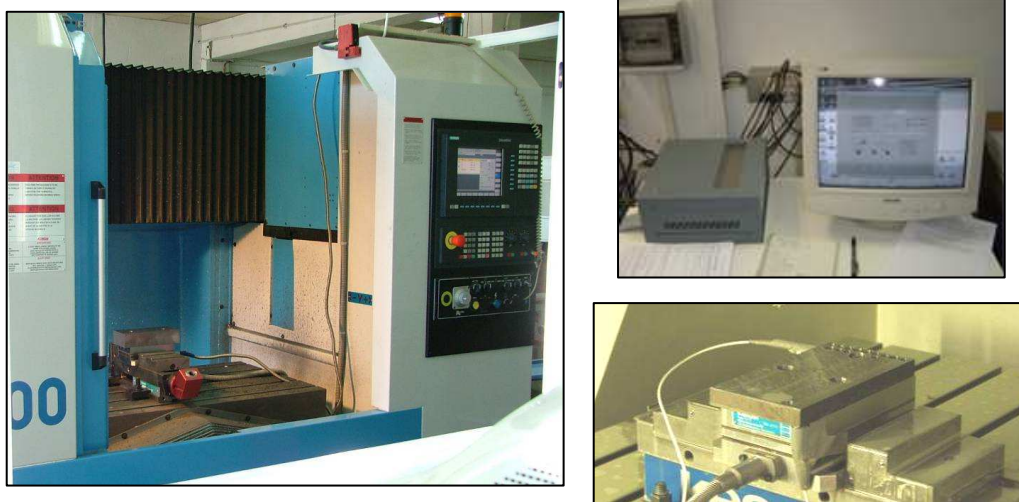


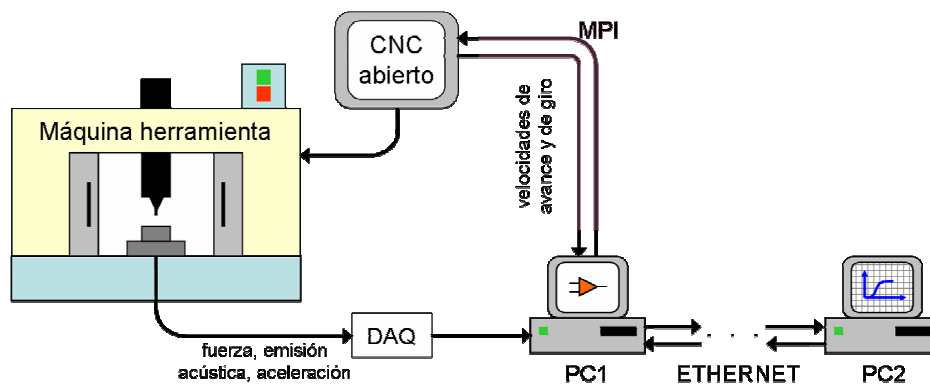
Figura 2.4. Componentes de la plataforma experimental

El CNC abierto de la máquina, registra en tiempo real todas las señales de los motores de los ejes de la máquina, como sus velocidades, posiciones, aceleraciones, par, etc., controlando a su vez variables del proceso como la velocidad de giro del cabezal, la velocidad de avance de la máquina, etc. Toda la información que maneja el CNC, se encuentra disponible a través de un enlace de red bi-direccional denominado MPI (*multi-point interface*), con una velocidad de transmisión de datos de 187.5 kbit/s, muy similar al bus de campo PROFIBUS de un uso

muy amplio en la industria en general. A través del bus MPI, también es posible cambiar los valores de referencia de las variables controladas por el CNC, como la velocidad de giro del cabezal y la velocidad en la dirección de avance de la máquina. Esto puede realizarse desde un ordenador remoto o mediante un sistema de control distribuido en red basado en comunicaciones *Ethernet*.

Además de las señales de fuerza y de las variables del CNC, se encuentran instalados sensores para monitorizar otras variables del proceso como la aceleración de vibraciones y emisión acústica. Estas señales son obtenidas mediante un sistema de adquisición de datos (DAQ) y pre-procesadas en una aplicación desarrollada en la interfaz LabView. Tanto las señales de fuerza como la aceleración son adquiridas hasta una frecuencia de muestreo de 50 kHz y la emisión acústica a una frecuencia de 1 MHz, mientras que las señales del CNC son muestreadas a la misma frecuencia de giro de la herramienta.

Como se puede intuir de la descripción anterior, la plataforma experimental reproduce un sistema de monitorización y control distribuido real pero a escala de laboratorio. La Figura 2.5 muestra una representación gráfica de todo el sistema.



**Figura 2.5. Diagrama de la plataforma experimental.**

Como ha podido apreciarse en la figura anterior, el sistema de monitorización y control distribuido se encuentra dividido en dos niveles de red diferentes (ver Figura 2.6). En el primer nivel definido por la red MPI (PROFIBUS), el ordenador PC1 contiene todo el sistema de adquisición y controla el acceso al CNC abierto. Este ordenador maneja todo el acceso al proceso físico, haciendo a su vez función de servidor de datos y señales del proceso. Empleando plataformas de desarrollo de aplicaciones en tiempo real, como LabView y Real-Time Workshop (RTW) de Matlab, en este ordenador es posible implementar algoritmos para la monitorización y/o control de los procesos de mecanizado.

El ordenador PC1 es, además, el encargado de realizar el enlace con el segundo nivel de red, definido por la red física *Ethernet*. En este nivel un ordenador PC2 puede ejecutar, tanto aplicaciones para monitorizar el proceso y estrategias para el control en red de las operaciones



de mecanizado, como algoritmos de optimización *on-line* de las estrategias de control que se ejecuten en cualquiera de los dos niveles de red. La transferencia de datos entre ambos ordenadores (PC1 y PC2) se realiza a través de la interfaz intermediaria RT-CORBA.

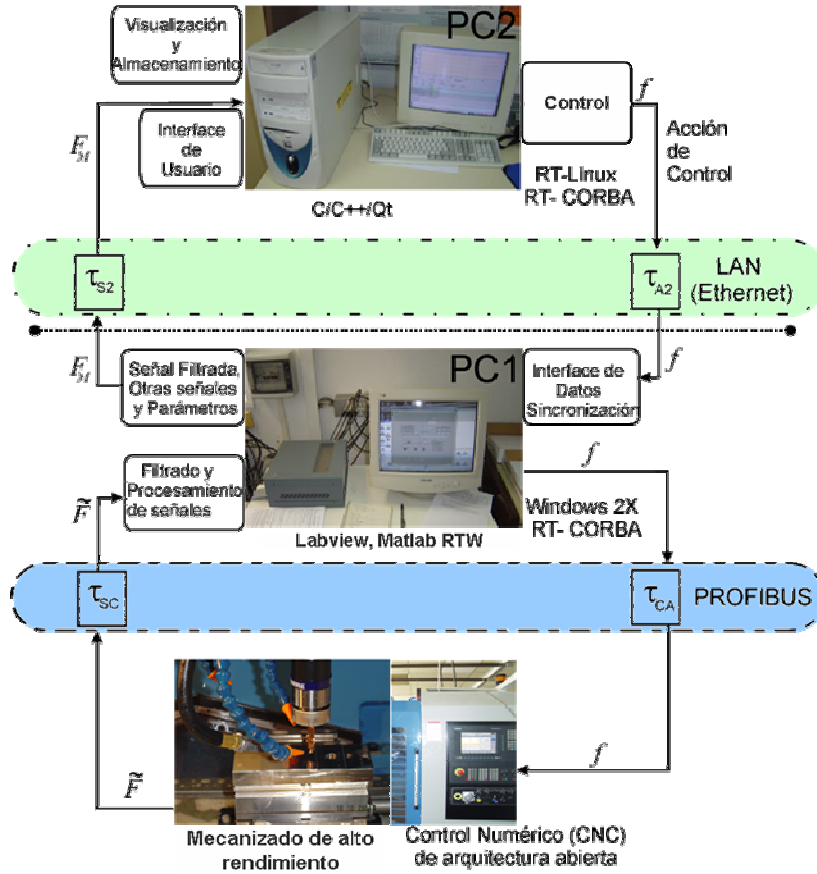


Figura 2.6. Niveles de red en la plataforma experimental de monitorización y control.

La existencia de diversos niveles de red, lleva asociado una serie de retardos de tiempo a la hora de realizar tareas de control o monitorización, que se deben tener en cuenta en la etapa de diseño de las diferentes estrategias. Estos retardos se adicionan a su vez al propio tiempo muerto intrínseco en los accionamientos electromecánicos que intervienen en las operaciones de mecanizado. Los retardos inducidos por la red usualmente no son conocidos *a priori* y, en general, son de difícil medición o estimación en línea. Un tratamiento formal acerca de los retardos inducidos por la red y el procedimiento para estimarlos puede ser encontrado en la literatura [131, 132].

El acceso al proceso desde el primer nivel de red implica un máximo retardo  $\tau_{p1}$  que incluye el retardo inducido por la red de este nivel  $\tau_{CA}$ , el retardo del sistema de adquisición  $\tau_{SC}$  y el tiempo muerto  $\tau_p$  del proceso (2.10). De la misma forma, el acceder al proceso

desde el segundo nivel de red, implicará un retardo adicional máximo  $\tau_{12}$  determinado por el tiempo de transmisión entre ambos ordenadores  $\tau_{S2}$  y  $\tau_{A2}$  (2.11).

$$\tau_{P1} = [\tau_{SC} + \tau_{CA} + \tau_P]_{MAX} \quad (2.10)$$

$$\tau_{P2} = [\tau_{P1} + \tau_{12}]_{MAX} = [\tau_{P1} + \tau_{S2} + \tau_{A2}]_{MAX} \quad (2.11)$$

## **2.2 Diseño e implementación de un modelo computacional para emular el comportamiento de variables representativas en procesos de fabricación. Análisis del caso particular de las fuerzas de corte en el proceso de fresado de alta velocidad**

En la última década los procedimientos de transformación de los metales han continuado centrando la atención de la comunidad científica internacional con numerosos trabajos dirigidos al desarrollo y perfeccionamiento de dichos métodos. El aprovechamiento de nuevas herramientas computacionales para el modelado asistido por ordenador, como es el caso de Matlab/Simulink, constituye una necesidad insoslayable para la mejora de estos procesos.

Dentro de los procesos de fabricación, los procesos de conformado por eliminación o arranque de material o, como se denominan comúnmente, procesos de mecanizado, incluyen algunos de los procesos productivos más intensamente utilizados. Por este motivo, algunas investigaciones apuntan hacia nuevos métodos que permitan alcanzar alto rendimiento y eficiencia en estos procesos, ya sea mediante un diseño más eficiente de las herramientas de corte empleadas, para reducir su desgaste [133] o para obtener un mejor acabado superficial [134].

De todas las variables involucradas en estos procesos, la fuerza de corte es una de las más significativas por la información relevante que aporta en la caracterización de estos procesos. Varios estudios realizados relacionan la fuerza de corte y por tanto los parámetros de corte, con el desgaste de la herramienta [135, 136] y el acabado superficial [137] en los procesos de mecanizado.

En esta sección se presenta la formalización e implementación en Matlab/Simulink, de un modelo matemático que relaciona las fuerzas y el par de corte, con los parámetros de corte y de la geometría generalizada de la herramienta en el conocido como proceso de fresado de alto rendimiento o fresado de alta velocidad (FAV). La elaboración del modelo de las fuerzas de corte aquí presentado tiene sus orígenes en trabajos precedentes donde se introducen modelos mecánicos y dinámicos que relacionan las fuerzas de corte con los parámetros de corte y la geometría de la herramienta.

El modelo que permite estimar las fuerzas de corte en el FAV, ha sido implementado con anterioridad en lenguajes potentes de cálculo. En un trabajo precedente se describe el

desarrollo del modelo en Matlab considerando solamente los tipos de fresa cilíndricas y de punta esféricas [135], también en otras plataformas de desarrollo como LabView pero solo para fresas de punta semiesférica [138]. El proceso de corte ha sido simulado, para la fresa de punta semiesférica, en lenguajes de alto nivel como C++, interactuando con herramientas para el modelado en 3D [139]. Además, el modelo para las fuerzas de corte en el mecanizado convencional fue introducido en trabajos previos, considerando la geometría generalizada de la herramienta [140] para los tipos de fresas helicoidales más utilizadas en la industria. Asimismo, para la fresa de punta semiesférica se han desarrollado modelos teniendo en cuenta, para el cálculo del espesor de la viruta, la influencia de la dinámica del proceso de corte [141] y considerando el movimiento de la herramienta en superficies 3D [141, 142].

El modelo aquí presentado, cuya implementación se ha realizado en una aplicación en Matlab/Simulink, a diferencia de trabajos previos incorpora la posibilidad de predecir las fuerzas de corte para diferentes tipos de herramientas, caracterizadas solamente por parámetros de su geometría. Además permite evaluar la influencia de la dinámica del proceso de corte en las fuerzas y el par desarrollado, producto de las vibraciones de la herramienta, y simular su comportamiento temporal considerando operaciones de mecanizado en superficies 3D.

El poder disponer de un método de cómputo eficiente para la predicción o estimación que, a partir de las condiciones de corte prefijadas por el tipo de material a mecanizar y la geometría de la herramienta empleada, sea capaz de predecir con cierto grado de precisión el posible comportamiento de las fuerzas y el par de corte, resulta de gran importancia. El comportamiento dinámico de dichas variables permitiría detectar posibles roturas en la herramienta de corte, estimar su desgaste, predecir la presencia de vibraciones y aportar información acerca del acabado superficial. Para ello resulta clave la obtención de un modelo que permita la simulación del proceso y la predicción del comportamiento de estas variables.

### 2.2.1 Modelo físico-matemático del proceso de fresado de alta velocidad

Un modelo del proceso de fresado de alta velocidad, permite estimar las fuerzas que se producen durante la interacción del filo de la herramienta de corte y el material a cortar, es decir, durante el proceso de corte o arranque de material. Para ello se determinan primeramente las fuerzas ( $dF$  [N]) en una sección infinitesimal del filo de corte ( $dS$  [mm]) de la herramienta y seguidamente se realiza una integración a lo largo de toda la zona de contacto. Las fuerzas que se generan (ver Figura 2.7) dependen, por una parte, de las características geométricas del filo de la herramienta y del material que lo compone ( $K_e$  [N/mm]) y, por otra parte, la dureza o composición química del material que se está mecanizando que a su vez determinará la fuerza específica o fuerza necesaria ( $K_c$  [N/mm<sup>2</sup>])

para que se produzca el arranque de una unidad de sección de material [143]. La ecuación (2.12) es una representación aproximada y generalizada para el cálculo de las componentes de fuerza que se producen. El área o sección de material ( $A[\text{mm}^2]$ ) que se desprende en forma de viruta, es proporcional a la relación entre la velocidad de avance de la herramienta con respecto a la pieza ( $f[\text{mm/min}]$ ) y su velocidad de giro ( $n[\text{l/min o rev/min}]$ ), es decir,  $A \propto f/n$ .

$$dF = K_e dS + K_c A \quad (2.12)$$

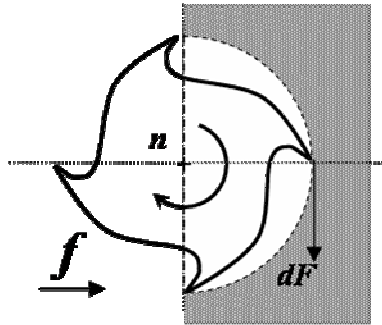


Figura 2.7. Esquema del proceso de FAV.

Por otra parte, la potencia específica o potencia requerida para que se produzca el arranque de una unidad de sección de material, depende de la fuerza y de la velocidad de corte o velocidad tangencial en el punto de corte,  $dP = dF V_c$ .

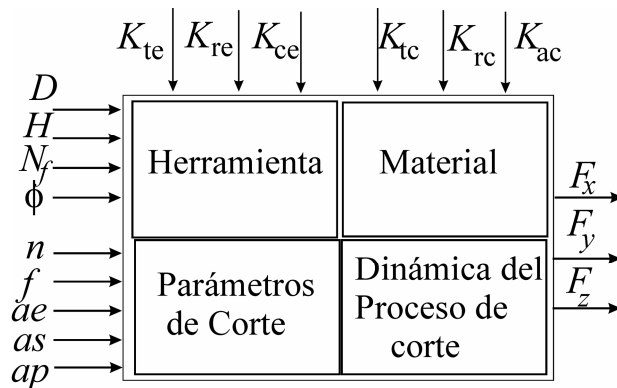


Figura 2.8. Diagrama del proceso de FAV visto como una caja negra.  
(Haber et ál., 2004 [135])

En general, es posible caracterizar el proceso de FAV a partir de las fuerzas de corte y, desde un enfoque sistémico, este se puede considerar como un sistema de múltiples entradas y múltiples salidas (MIMO) [135]. Dicho sistema involucra parámetros y variables que definen

la herramienta, el material y la dinámica inherente al proceso de corte, tal y como se muestra en la Figura 2.8.

A continuación se detallan todos los pasos para la obtención del modelo físico del proceso de fresado. Por claridad, seguidamente se resume la nomenclatura de las variables empleadas en las ecuaciones:

$D, R, R_r, R_z$	Parámetros radiales de la herramienta [mm].
$\alpha, \beta$	Parámetros angulares de la herramienta [mm].
$H$	Altura total del filo de corte de la herramienta [mm].
$i_0$	Ángulo nominal de la hélice [grados].
$N_f$	Número de dientes.
$f$	Avance de la máquina [mm/min].
$V_c$	Velocidad de corte [m/s].
$P_c$	Potencia de corte [W].
$ap$	Profundidad axial de corte [mm].
$ae$	Profundidad radial de corte [mm].
$K_{tc}, K_{rc}, K_{ac}$	Constantes de las fuerzas de corte en sentido tangencial, radial y axial referidas al material [N/mm <sup>2</sup> ].
$K_{te}, K_{re}, K_{ae}$	Constantes en sentido tangencial, radial y axial de la herramienta [N/mm].
$F_x, F_y, F_z$	Fuerzas a lo largo de los ejes X, Y, Z [N].
$F_{total}$	Fuerza resultante de las fuerzas en los tres ejes [N].
$\bar{F}_e, \bar{F}_t$	Esfuerzo de corte medio experimental y estimado por el modelo.
$T(\phi)$	Par resultante [Nm].
$r(z)$	Coordenada radial de un punto del filo de corte.
$P$	Un punto en el filo de corte.
$\psi(z)$	Ángulo radial de retraso.
$\kappa(z)$	Ángulo de inmersión axial.
$\phi_j(z)$	Ángulo de inmersión para el diente $j$ a la profundidad de corte $z$ .
$\phi$	Ángulo de rotación del filo de la fresa.
$\phi_p$	Paso angular entre dientes.
$h(\phi, z)$	Espesor de la viruta.
$h_r(\phi, z)$	Espesor radial de la viruta.
$n$	Velocidad del cabezal en [rpm] $n = (V_c \cdot 1000) / (D \cdot \pi \cdot 60)$ .
$s_{tj}$	Avance por diente en mm/rev $s_{tj} = f / (n \cdot N_f)$ .
$\phi_{st}$	Ángulo de inmersión radial en el punto de inicio.
$\phi_{ex}$	Ángulo de inmersión radial en el punto final.
$i(z)$	Ángulo de la hélice en función de la altura.

$dF_r, dF_t, dF_a$	Diferencial de fuerzas en los sentidos radial, tangencial y axial.
$dz$	Altura diferencial de un segmento de viruta.
$dS$	Sección infinitesimal del filo de corte de la herramienta.

### Geometría generalizada de la herramienta

El modelo geométrico a lo largo del filo de corte helicoidal requiere de un análisis cinemático y dinámico del proceso de corte. La predicción de las fuerzas de corte necesita de un sistema de coordenadas, del ángulo de la hélice y la distancia angular de un punto en el filo de corte [135, 140]. Este punto se encuentra a una profundidad axial de corte  $a_p$  en la dirección del eje Z, a una distancia radial  $r(z)$  en el plano XY, con un ángulo de inmersión axial  $\kappa(z)$  y un ángulo radial de retraso de  $\psi(z)$  (Figura 2.9).

El modelo matemático impuesto para el borde de corte, considera que el filo se divide en pequeños incrementos donde los coeficientes de corte pueden variar para cada localización. Se considera como punto de referencia inicial al primer filo de corte de la herramienta ( $j = 1$ ), y su ángulo de rotación cuando  $z = 0$  es  $\phi$ . El ángulo de inmersión radial para el filo de corte  $j$  en cierta posición axial  $z$  queda expresado como:

$$\phi_j(z) = \phi + (j-1)\phi_p - \psi(z) \quad (2.13)$$

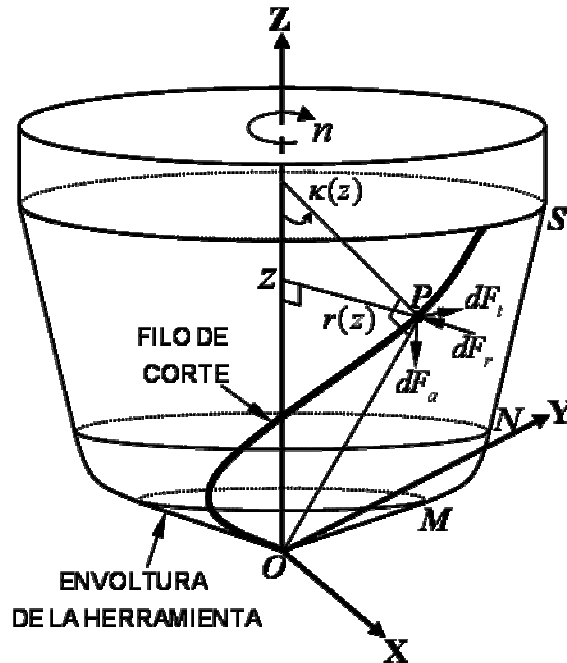


Figura 2.9. Modelo geométrico generalizado de una fresa.

El ángulo de retraso  $\psi(z)$  aparece debido al ángulo de la hélice  $i(z)$ . En el modelo generalizado para la geometría de una fresa con dientes helicoidales, el diámetro de la herramienta puede ser diferente a lo largo de ésta, dependiendo de la forma de la herramienta (cilíndrica plana, semiesférica, de bola, cónica, etc.).

Basándose en el modelo generalizado de la herramienta introducido en [140], ésta se describe por siete parámetros:  $D, R, R_r, R_z, \alpha, \beta, H$  (Figura 2.10). Una gran variedad de formas geométricas de herramientas pueden definirse a través de estos parámetros de acuerdo con el trabajo anteriormente citado.

La geometría de la herramienta se divide en tres zonas, el cono  $OM$ , el arco  $MN$  y la de estrechamiento  $NS$ . El punto  $M$  es la intersección de la línea  $OM$  y el arco  $MN$ , el punto  $N$  es la intersección del arco  $MN$  y la línea  $NS$ .

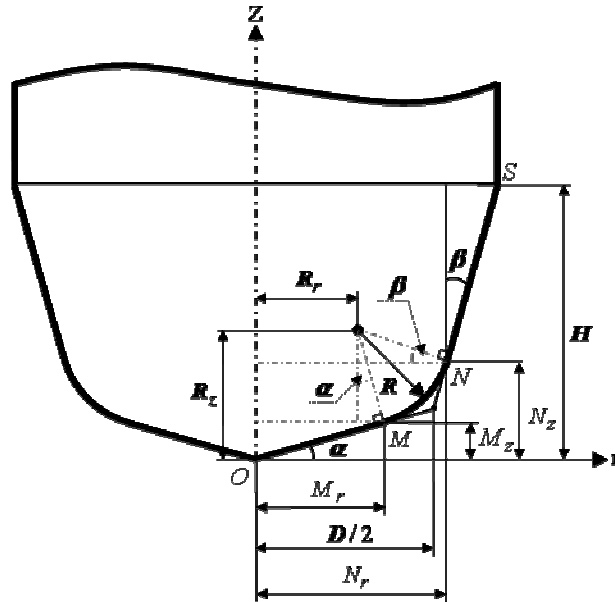


Figura 2.10. Geometría general de la herramienta.

Las distancias radiales y axiales desde los puntos  $M$  y  $N$  hasta el eje de la herramienta y el extremo de ésta, se definen a partir de las relaciones geométricas de los parámetros generales tal y como se puede observar en la Figura 2.10. Asumiendo que la hélice es continua, es decir, las zonas son tangentes entre sí en su intersección, las distancias para cada punto quedan definidas de la siguiente forma:

$$M_r = R_r + R \sin \alpha, \quad M_z = M_r \tan \alpha \quad (2.14)$$

$$\begin{aligned} N_r &= R_r + R \cos \beta \\ N_z &= R_z - R \sin \beta \end{aligned} \quad (2.15)$$

Las ecuaciones anteriores son válidas sólo en el caso de que  $0 \leq \alpha < \pi/2$  y  $\beta < \pi/2$  y su planteamiento es más sencillo de implementar que el introducido en la referencia [140], evitando además una posible indefinición de  $M_r$  y  $N_z$  para el caso de que  $R=0$ . Esta reformulación constituye una contribución al modelado de este proceso.

La distancia radial  $r(z)$ , el ángulo de inmersión axial  $\kappa(z)$ , el ángulo radial de retraso  $\psi(z)$ , el ángulo de hélice  $i(z)$  y la longitud infinitesimal  $dS$  del filo en el punto de corte  $P$ , se definen en [140], según la distancia axial  $z$  del punto para cada una de las zonas de la geometría de la herramienta.

### ***Espesor radial instantáneo de la viruta***

El espesor radial de la viruta  $h_r(\phi_j, z)$  presenta componentes estáticas,  $h_{r\_st}(\phi_j, z)$ , y dinámicas,  $h_{r\_din}(\phi_j, z)$  [144]. Éste varía dinámicamente debido a las vibraciones de la herramienta. Considerando que estas vibraciones se producen en el plano XY, el espesor dinámico depende de la vibración del diente que está cortando actualmente y de la vibración del diente en el periodo anterior  $\tau$  [seg] [145].

$$h_r(\phi_j, z) = h_{r\_st}(\phi_j, z) + h_{r\_din}(\phi_j, z) \quad (2.16)$$

$$h_{r\_din}(\phi_j, z) = \Delta x(t) \sin \phi_j(t, z) - \Delta y(t) \cos \phi_j(t, z) \quad (2.17)$$

donde  $\Delta x(t)$  y  $\Delta y(t)$  son las variaciones en la posición de la herramienta debido a las vibraciones y se definen como,

$$\begin{aligned} \Delta x(t) &= x(t) - x(t - \tau) \\ \Delta y(t) &= y(t) - y(t - \tau), \quad \tau = \frac{60}{n N_f} \left[ \frac{\text{seg} / \text{min}}{\text{rev} / \text{min}} \right] \end{aligned} \quad (2.18)$$

Considerando que la operación de fresado se realiza en 3D (Figura 2.11), el espesor radial estático de la viruta  $h_{r\_st}(\phi_j, z)$  [141, 142] se define, considerando la geometría general de la herramienta, como,

$$h_{r\_st}(\phi_j, z) = s_{ij} \frac{dr(z)}{dz} \sin \Phi + s_{ij} \sin \phi_j(t, z) \cos \Phi \quad (2.19)$$

donde  $\Phi$ , en este caso mayor que 0, es el ángulo de la pendiente de la superficie a mecanizar. Si la operación de fresado se estuviera realizando sólo en el plano XY, es decir  $\Phi=0$ , entonces se tendría como ecuación para la determinación del espesor radial estático de la viruta,  $h_{r\_st}(\phi_j, z) = s_{ij} \sin \phi_j(t, z)$  [146, 147].



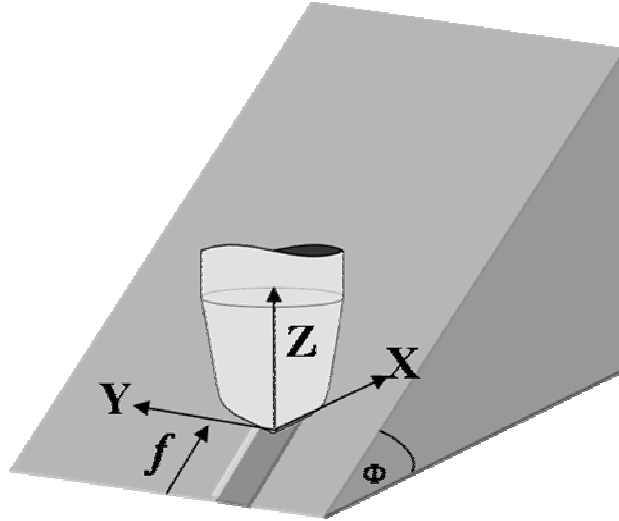


Figura 2.11. Operación de fresado en 3D con el ángulo de elevación de la superficie  $\Phi$ .

### Modelo físico de las fuerzas y el par de corte

Para determinar las fuerzas de corte implicadas en una operación de fresado [135, 140, 148, 149] se determinan los diferenciales de fuerza en el sentido tangencial ( $dF_t$ ), radial ( $dF_r$ ) y axial ( $dF_a$ ) (Figura 2.9), que actúan sobre un elemento infinitesimal del filo de corte de la herramienta (punto  $P$ ) y seguidamente éstas son transformadas geoméricamente al sistema de coordenadas cartesianas  $XYZ$ . Integrando los diferenciales de fuerzas en el nuevo sistema de coordenadas en todo el filo de corte y sumando las fuerzas en todos los dientes de la herramienta, la ecuación general para determinar las fuerzas totales  $\mathbf{F}_{xyz}(\phi) = [F_x(\phi) \ F_y(\phi) \ F_z(\phi)]^T$  según el ángulo de giro  $\phi$  se puede expresar en la forma matricial-vectorial como:

$$\mathbf{F}_{xyz}(\phi) = \sum_{j=1}^{N_f} \left\{ \int_{z_1}^{z_2} [\mathbf{A}(\phi_j, \kappa) \mathbf{K}_e \frac{dS}{dz} + \mathbf{A}(\phi_j, \kappa) \mathbf{K}_e h_r(\phi_j, z)] dz \right\} \quad (2.20)$$

donde  $h_r(\phi_j, z)$  es el espesor radial de la viruta,  $\mathbf{K}_e = [K_{re} \ K_{te} \ K_{ae}]^T$  y  $\mathbf{K}_c = [K_{rc} \ K_{tc} \ K_{ac}]^T$  son los coeficientes de corte, los cuales pueden ser determinados de forma experimental para un tipo dado de herramienta y material y  $\mathbf{A}(\phi_j, \kappa)$  es la matriz de transformación geométrica, la cual se define como,

$$\mathbf{A}(\phi_j, \kappa) = \begin{bmatrix} -\sin \phi_j \sin \kappa & -\cos \phi_j & -\sin \phi_j \cos \kappa \\ -\cos \phi_j \sin \kappa & \sin \phi_j & -\cos \phi_j \cos \kappa \\ -\cos \kappa & 0 & -\sin \kappa \end{bmatrix} \quad (2.21)$$

Debido a la intermitencia en el proceso de corte la ecuación (2.20) sólo es válida si se cumple que  $\phi_{st} \leq \phi \leq \phi_{ex}$ , es decir, que el labio de la herramienta se encuentra insertado dentro de la pieza, de otra forma las tres fuerzas se consideran nulas para ese labio.  $\phi_{st}$  y  $\phi_{ex}$  son los ángulos de inmersión radial de entrada y salida respectivamente a la pieza.

En la ecuación (2.20),  $z_1$  y  $z_2$  son los límites de integración de la zona de contacto en cada momento del corte y pueden calcularse a partir del modelo geométrico generalizado de la fresa,  $dz$  es el diferencial de la profundidad de corte axial y el ángulo de rotación  $\phi$  está definido por:

$$\phi = \Omega \cdot t \quad (2.22)$$

donde  $\Omega$  es la velocidad del cabezal en (*rad* / *s*).

Del vector de fuerzas  $\mathbf{F}_{xyz}(\phi)$  se determina su valor resultante como,

$$F_{total}(\phi) = \|\mathbf{F}_{xyz}(\phi)\| = \sqrt{F_x(\phi)^2 + F_y(\phi)^2 + F_z(\phi)^2} \quad (2.23)$$

A partir del diferencial de fuerza tangencial  $dF_t$  se puede determinar el par de corte, quedando expresado de la siguiente forma:

$$\begin{aligned} T(\phi) &= \sum_{j=1}^{N_f} T(\phi_j) = \sum_{j=1}^{N_f} \int_{z_1}^{z_2} r(z) dF_t(\phi_j, z) = \\ &= \sum_{j=1}^{N_f} \int_{z_1}^{z_2} r(z) \left[ K_{te} \frac{dS}{dz} + K_{tc} h_r(\phi_j, z) \right] dz \end{aligned} \quad (2.24)$$

### 2.2.2 Descripción del modelo implementado en Simulink

A partir de los modelos matemáticos se implementaron los algoritmos en Matlab/Simulink. La aplicación desarrollada en Simulink permite de forma muy sencilla a través de una interfaz de usuario introducir los parámetros de la geometría de la herramienta y de corte (Figura 2.12 y Figura 2.13).

La Figura 2.12 muestra el diagrama de bloques principal (función principal de la aplicación) del modelo físico-matemático de las fuerzas y el par de corte implementado en Simulink. En él se puede observar, mediante gráficas, el comportamiento de las fuerzas, cartesianas y resultante, y el par de corte. La ecuación (2.23), correspondiente a la fuerza resultante,  $F_{total}(\phi)$ , aparece implementada explícitamente en este diagrama. En la parte izquierda de la Figura 2.12 se muestra el árbol de dependencias entre los diferentes diagramas

o subsistemas (funciones de la aplicación) que conforman el modelo (ver también Figura 2.14).

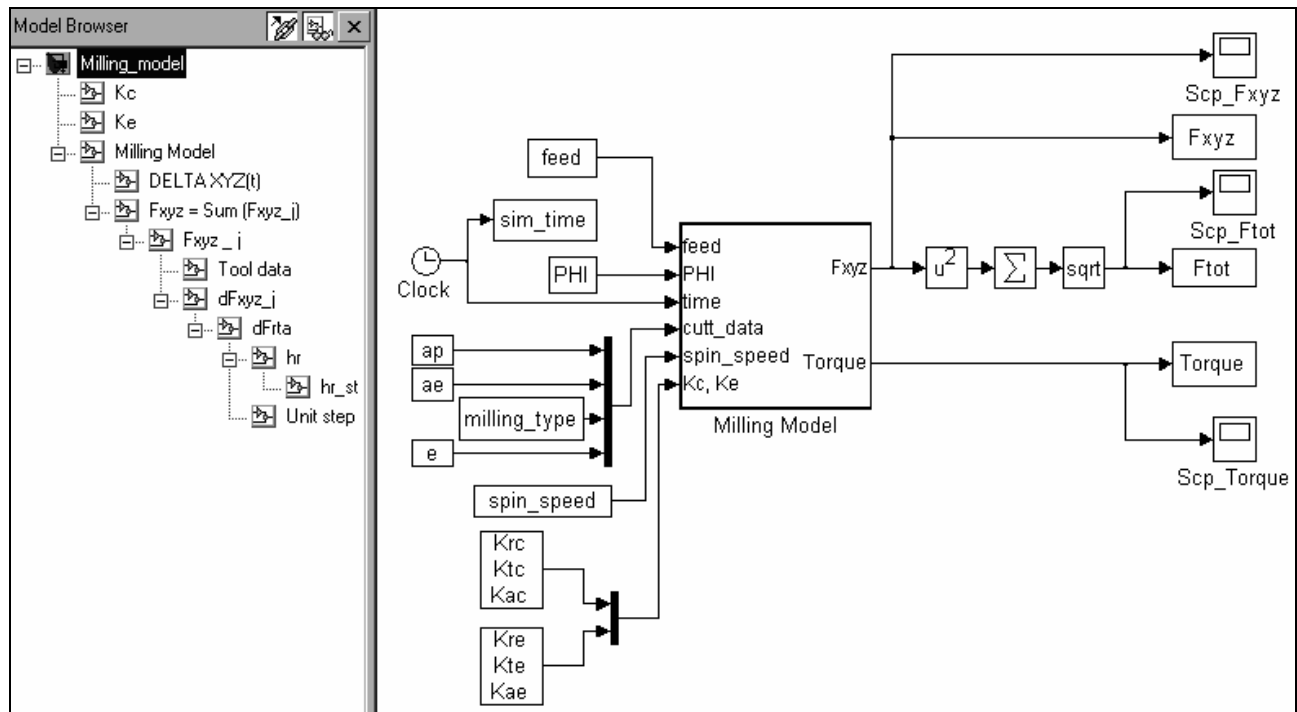


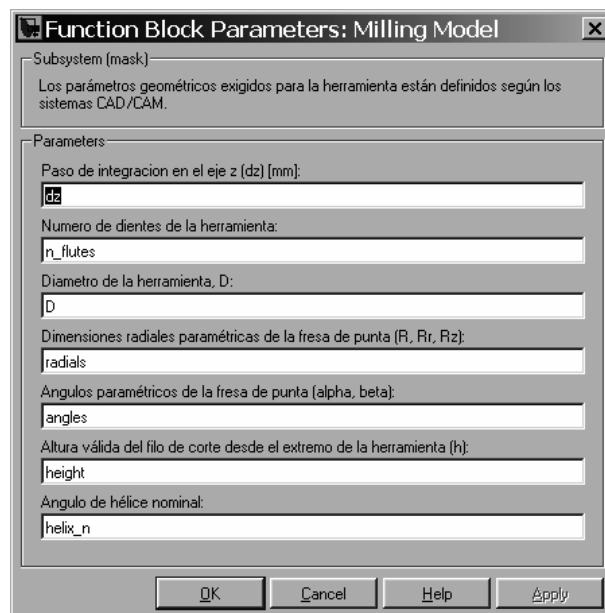
Figura 2.12. Diagrama de bloques del modelo matemático de las fuerzas y el par de corte en SIMULINK.

El diagrama de la Figura 2.12 contiene también las entradas y salidas del modelo, las cuales se describen a continuación:

<b>Entradas</b>	feed	Avance de la máquina [mm/min].
	PHI	Ángulo de elevación de la superficie que se va a mecanizar en cada instante de tiempo.
	time	Tiempo de la simulación.
	cutt_data	Datos de corte.
	ap	Profundidad axial de corte [mm].
	ae	Profundidad radial de corte [mm].
	milling_type	Especifica el tipo de operación de fresado, ya sea contorneado o planeado.
	e	Excentricidad entre la herramienta y la pieza [mm].
	spin_speed	Velocidad del cabezal [rpm].
	Kc, Ke	Vectores de las constantes de corte del material y la herramienta.
<b>Salidas</b>	Fxyz	Vector de fuerzas cartesianas.

$F_{tot}$	<i>Fuerza resultante o norma del vector de fuerzas cartesianas [N].</i>
Torque	<i>Par de corte resultante [Nm].</i>

En la interfaz de usuario utilizada para la entrada de los parámetros de la geometría de la herramienta (Figura 2.13) es posible introducir parámetros tales como el paso de integración en el eje  $z$  ( $dz$ ), el número de dientes de la herramienta, el diámetro de la herramienta, los parámetros radiales de la herramienta ( $R$ ,  $R_r$ ,  $R_z$ , ver Figura 2.10), los ángulos de la geometría de la herramienta ( $\alpha$ ,  $\beta$ ), la altura total del filo de corte ( $H$ ) y el ángulo de hélice nominal. El modelo utiliza el método numérico de Euler para la integración de las fuerzas de corte a lo largo del filo de cada labio, a la profundidad axial que se está cortando, aunque Simulink deja abierta la posibilidad a utilizar otros métodos como Runge-Kutta de cuarto orden.



**Figura 2.13.** Interfaz de usuario del modelo en SIMULINK para la entrada de parámetros de la geometría de la herramienta.

La implementación del modelo se realizó con cierto nivel de encapsulamiento y de forma estructurada, garantizando que el desarrollo y el manejo de cada uno de los diagramas o subsistemas que lo componen, fuesen lo menos complejo posible y computacionalmente más eficiente. En la Figura 2.14 se muestran las relaciones o dependencias entre cada uno de los subsistemas del modelo tal y como fueron implementados en Simulink.

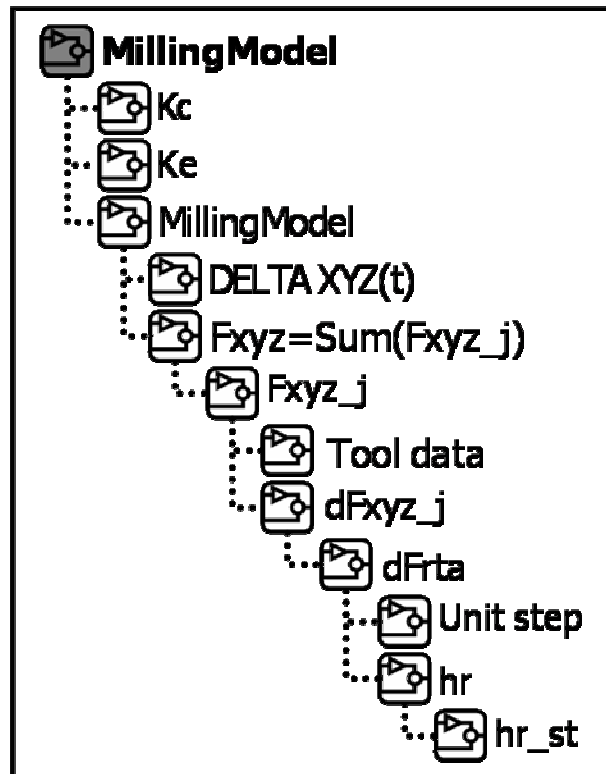


Figura 2.14. Árbol de dependencias entre los diagramas o subsistemas que conforman el modelo desarrollado.

A continuación se relacionan cada uno de los subsistemas con una breve descripción de su funcionamiento:

<b>MillingModel</b>	Constituye el diagrama principal del modelo, en él se tiene acceso a la interfaz gráfica para introducir los parámetros de la geometría de la herramienta, los parámetros de corte y las constantes de corte, además permite observar mediante gráficas las fuerzas y el par de corte. Contiene el subsistema nombrado MillingModel, donde está implementado todo el modelo del proceso de fresado.
Kc	Se encuentra dentro del diagrama principal, es el que implementa la interfaz gráfica para introducir las constantes de corte $K_{tc}$ , $K_{rc}$ , $K_{ac}$ y las almacena en variables del modelo.
Ke	Este diagrama se encuentra dentro del diagrama principal, implementando la interfaz gráfica para introducir las constantes de corte $K_{tc}$ , $K_{rc}$ , $K_{ac}$ , almacenándolas en variables del modelo.
MillingModel	En este subsistema se implementa el cálculo del avance por diente, $s_{ij}$ . Además se calcula el valor del ángulo inmersión radial, $\phi_j$ , de cada diente, en dependencia del tiempo y del paso angular entre dientes $\phi_p$ . Igualmente se realiza el cálculo de los ángulos de inicio y fin de corte, $\phi_{st}$ y $\phi_{ex}$ , a partir de los parámetros de corte y del tiempo actual, implementando las ecuaciones de la cinemática de estos ángulos introducida en la

	sección anterior, también se determinan los límites de integración $z_1$ y $z_2$ a partir de la profundidad axial de corte, $ap$ . Se realiza una llamada a los subsistemas $F_{xyz} = \text{Sum}(F_{xyz\_j})$ y $\text{DELTA XYZ}(t)$ .
DELTA XYZ(t)	Con este diagrama de bloques se determina la variación de la posición de la herramienta debido a las vibraciones (ecuación (2.18))
$F_{xyz} = \text{Sum}(F_{xyz\_j})$	Dentro de este diagrama de bloques se determina el valor del vector de fuerzas totales de cada eje cartesiano, $\mathbf{F}_{xyz}(\phi)$ (ecuación (2.20)), a partir de las fuerzas por cada diente y el par de corte total $T(\phi)$ (ecuación (2.24)). Además determina, a partir del paso de integración $dz$ en el eje $z$ y los límites de integración $z_1$ y $z_2$ , el número total de incrementos en los que se divide el filo de la herramienta.
$F_{xyz\_j}$	Este subsistema realiza la integración a lo largo del filo de la herramienta mediante el método numérico de integración Euler, determinando el vector de fuerzas y del par de corte por cada diente.
Tool data	La llamada a este diagrama de bloques se realiza dentro del subsistema $F_{xyz\_j}$ e implementa, a partir de los parámetros de la geometría de la herramienta y de la distancia axial $z$ del punto $P$ de corte, el cálculo de la distancia radial $r(z)$ , su diferencial respecto al eje $z$ , $dr(z)/dz$ , el ángulo de inmersión axial $\kappa(z)$ , el ángulo radial de retraso $\psi(z)$ y la longitud infinitesimal $dS$ .
$dF_{xyz\_j}$	Este subsistema realiza la transformación geométrica de las fuerzas en el sentido tangencial, radial y axial al sistema de coordenadas $\mathbf{XYZ}$ .
$dF_{rta}$	En este diagrama se calcula el vector de fuerzas en el sentido tangencial, radial y axial, a partir del espesor de viruta, los vectores de constantes de corte, $\mathbf{K}_c$ y $\mathbf{K}_e$ , el avance por diente $s_{ij}$ y la longitud infinitesimal del filo de la herramienta $dS$ . En el cálculo se tiene en cuenta la intermitencia en el proceso de corte.
Unit step	Este subsistema determina la continuidad de las fuerzas de corte a partir del ángulo de giro $\phi_j$ de cada labio de la herramienta y los ángulos de inicio y fin de corte, $\phi_{st}$ y $\phi_{ex}$ .
hr	Diagrama de bloques que implementa el cálculo del espesor radial de la viruta $h_r(\phi_j, z)$ (ecuaciones (2.16) y (2.17)).
hr_st	Este subsistema es el encargado de calcular la componente estática del espesor radial de la viruta, $h_{r\_st}(\phi_j, z)$ (ecuación (2.19)).

En la Figura 2.15 se muestran otros diagramas de bloques que forman el modelo, mostrándose también a qué subsistema pertenece dentro del árbol de dependencias del

modelo. En la Figura 2.15a) se muestra el diagrama de bloques correspondiente al cálculo de los diferenciales de fuerza en los ejes cartesianos, representados en este caso como  $[dF_x; dF_y; dF_z]$  y el diferencial de par de corte,  $dTorque$ . En el diagrama las fuerzas se calculan a través de la matriz de transformación geométrica, representada como  $T\_RTA\_XYZ$ , aplicada al vector de diferenciales de fuerzas radiales, tangenciales y axiales, determinadas en este caso por el bloque de llamada de subsistema nombrado como  $dFrta$ . En la Figura 2.15b) aparece el diagrama correspondiente al cálculo del espesor radial estático de la viruta denominado como  $hr\_st$ , ecuación (2.19), calculado a partir del avance por diente,  $St_j$ , el diferencial del radio de la herramienta respecto a la altura,  $dR/dz$ , el ángulo de elevación de la superficie a mecanizar,  $PHI$ , y el ángulo de inmersión radial correspondiente al diente  $j$ ,  $phi\_j$ .

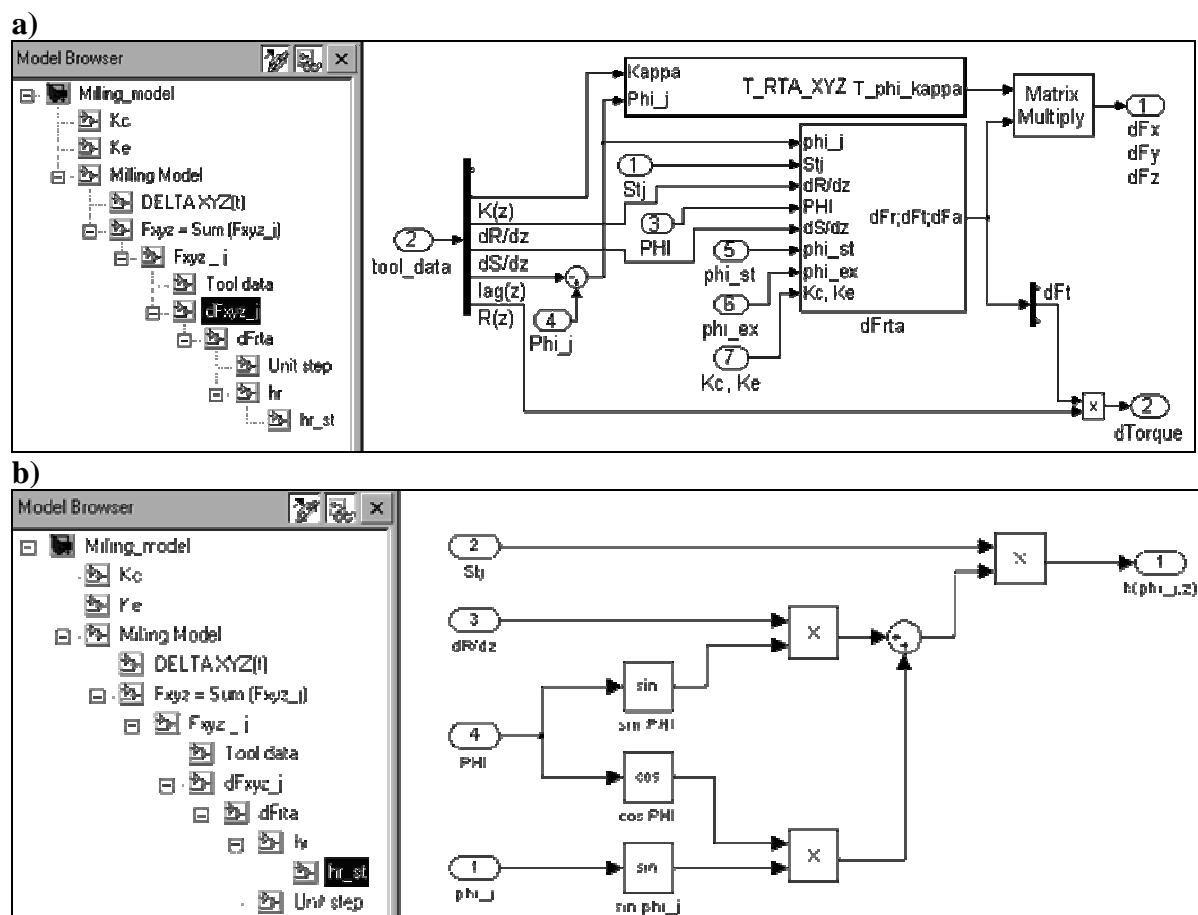
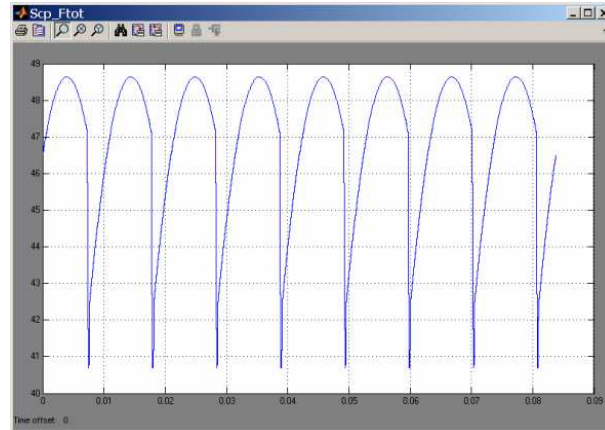


Figura 2.15. Otros diagramas de bloques que componen el modelo implementado en SIMULINK.

a) Diagrama que implementa el cálculo de los diferenciales de fuerzas cartesianas y el diferencial de par de corte, b) diagrama de bloques para el cálculo del espesor radial estático de la viruta.

El modelo permite también, según los parámetros introducidos, la evaluación visual del comportamiento teórico de las fuerzas de corte a través de gráficas, tanto resultante (Figura 2.16) como en cada eje cartesiano y el par resultante.



**Figura 2.16.** Gráfica de fuerza de corte resultante obtenida del modelo implementado en SIMULINK para ciertos parámetros de entrada.

### 2.2.3 Validación experimental del modelo de fuerzas de corte

Con vistas a evaluar el comportamiento del modelo implementado se utilizaron datos experimentales obtenidos de la plataforma experimental descrita con anterioridad. Se adquirieron las señales de fuerza en los tres ejes cartesianos,  $F_x$ ,  $F_y$  y  $F_z$ , a una frecuencia de muestreo de 40kHz. Se utilizó como material de trabajo piezas de aluminio de una dureza Brinell de 65 HB5 para un primer experimento y de 145 HB5 para otros tres restantes.

En el estudio realizado, en la simulación y en los ensayos reales, se consideraron las siguientes condiciones de corte para operaciones de fresado a alta velocidad:

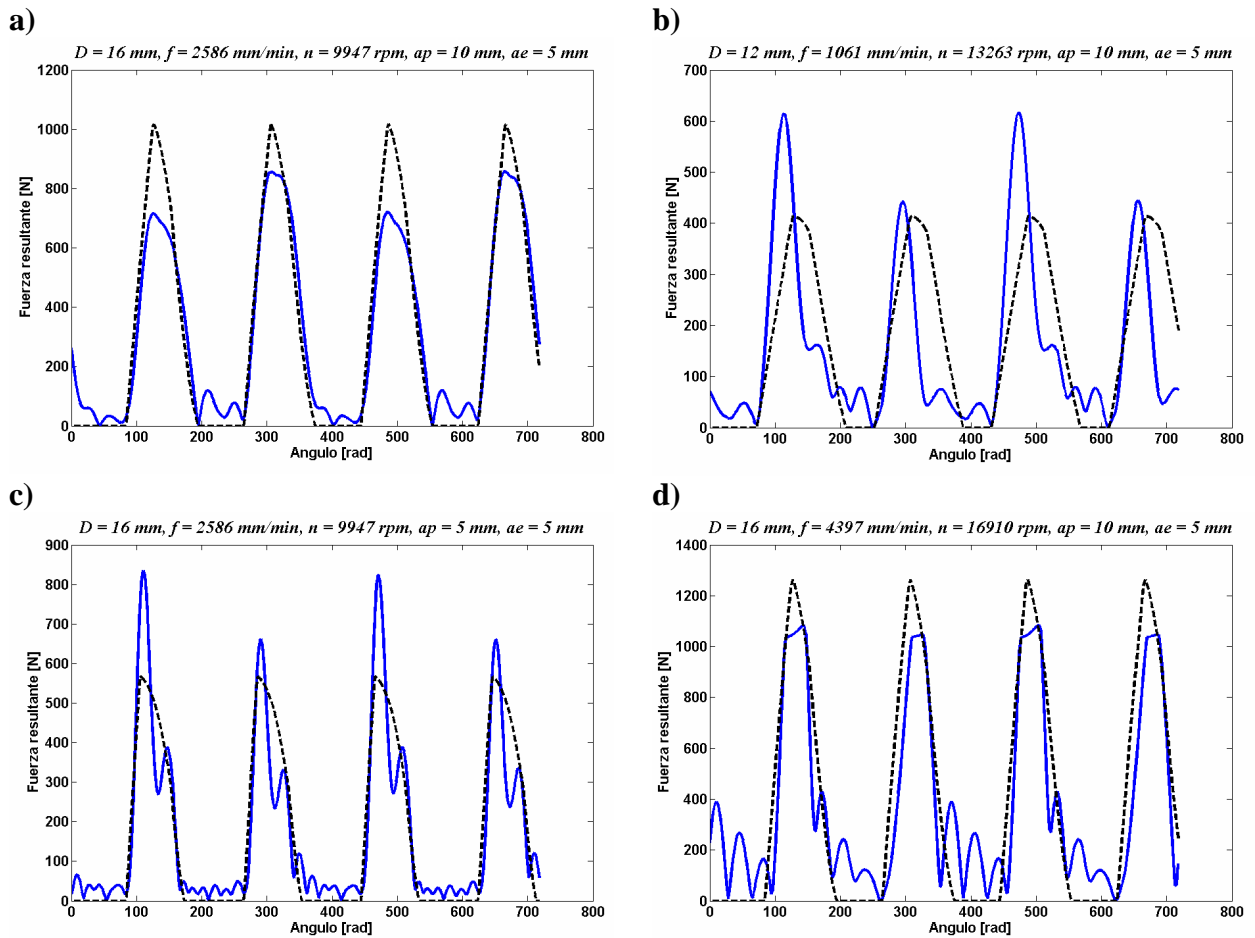
- $V_c=500$  m/min,  $n=9947$  rpm,  $f=2586$  mm/min,  $a_p= 10$  mm,  $a_e= 5$  mm,  $i_0=25^\circ$ ,  $H=20.0$  mm,  $D= 16.0$  mm.
- $V_c=500$  m/min,  $n= 13263$  rpm,  $f= 1061$  mm/min,  $a_p= 10$  mm,  $a_e= 5$  mm,  $i_0=25^\circ$ ,  $H=16.0$  mm,  $D= 12.0$  mm.
- $V_c=500$  m/min,  $n= 9947$  rpm,  $f= 2586$  mm/min,  $a_p= 5$  mm,  $a_e= 5$  mm,  $i_0=25^\circ$ ,  $H=20.0$  mm,  $D= 16.0$  mm.
- $V_c=850$  m/min,  $n= 16910$  rpm,  $f= 4397$  mm/min,  $a_p= 10$  mm,  $a_e= 5$  mm,  $i_0=25^\circ$ ,  $H=20.0$  mm,  $D= 16.0$  mm.

Las constantes de corte utilizadas en la simulación y en la validación experimental se ajustaron experimentalmente para cada conjunto material-herramienta y según las condiciones de corte. Para cada caso estas constantes tienen los siguientes valores:

- $K_{tc}= 568.21$  N/mm<sup>2</sup>,  $K_{rc}= 416.53$  N/mm<sup>2</sup>,  $K_{ac}= 61.85$  N/mm<sup>2</sup>,  $K_{te}= 11.26$  N/mm,  $K_{re}= 14.20$  N/mm,  $K_{ae}= 2.38$  N/mm.



- b)  $K_{tc} = 496.81 \text{ N/mm}^2$ ,  $K_{rc} = 364.19 \text{ N/mm}^2$ ,  $K_{ac} = 54.08 \text{ N/mm}^2$ ,  $K_{te} = 9.84 \text{ N/mm}$ ,  $K_{re} = 12.41 \text{ N/mm}$ ,  $K_{ae} = 2.08 \text{ N/mm}$ .
- c)  $K_{tc} = 593.16 \text{ N/mm}^2$ ,  $K_{rc} = 434.81 \text{ N/mm}^2$ ,  $K_{ac} = 64.57 \text{ N/mm}^2$ ,  $K_{te} = 11.75 \text{ N/mm}$ ,  $K_{re} = 14.82 \text{ N/mm}$ ,  $K_{ae} = 2.48 \text{ N/mm}$ .
- d)  $K_{tc} = 707.33 \text{ N/mm}^2$ ,  $K_{rc} = 518.51 \text{ N/mm}^2$ ,  $K_{ac} = 77 \text{ N/mm}^2$ ,  $K_{te} = 14.01 \text{ N/mm}$ ,  $K_{re} = 17.67 \text{ N/mm}$ ,  $K_{ae} = 2.96 \text{ N/mm}$ .



**Figura 2.17.** Fuerza de corte resultante experimental (línea continua) y obtenida por el modelo para diferentes condiciones de corte (línea discontinua).

Con el propósito de validar el modelo desarrollado, se seleccionaron dos fresas cilíndricas planas modelo Sandvik R216.32-12025-AP16A H10F para el segundo experimento (b) y modelo R216.32-16025-AP20A H10F para los restantes (a, c, d). La primera es de 12 mm de diámetro y profundidad de corte máxima de 16 mm y la segunda herramienta tiene un diámetro de 16 mm y una profundidad de corte máxima de 20 mm. Según datos del fabricante, ambas herramientas son para operaciones de ranurado, presentan función de taladrado, tienen dos dientes, un ángulo de hélice de  $25^\circ$  y están hechas de un metal duro sin

recubrimiento, adecuadas para el fresado de aleaciones de aluminio con aristas de corte “agudizadas”. Las probetas fueron mecanizadas según un patrón en círculo.

En la Figura 2.17 se muestra el comportamiento real (experimental) de la fuerza de corte resultante para cada uno de los cuatro casos analizados (línea continua). En línea discontinua aparece representada la respuesta del modelo (simulación) según las condiciones de corte de cada uno de los experimentos. Se aprecia en todos los casos que el comportamiento del esfuerzo de corte resultante estimado se acerca bastante al comportamiento de los valores experimentales.

**Tabla 2.1. Comparación de las fuerzas de corte obtenidas experimentalmente y mediante el modelo.**

Experimento	$\bar{F}_e$ [N]	$\bar{F}_t$ [N]	$AE [\%] = \left  \frac{(\bar{F}_e - \bar{F}_t)100}{\bar{F}_e} \right $
a	319.38	324.63	1.64
b	158.84	173.13	8.99
c	175.15	171.08	2.32
d	386.02	403.81	4.61

Para evaluar el desempeño del modelo, se utilizaron varias cifras de mérito o índices de comportamiento para ambos experimentos. De esta manera se pueden comparar las fuerzas de corte obtenidas a través del modelo y las experimentales. En la Tabla 2.1 se muestra para cada experimento la fuerza resultante media obtenida experimentalmente,  $\bar{F}_e$ , y la estimada teóricamente por el modelo,  $\bar{F}_t$ , para estos dos valores se determinó el error absoluto relativo ( $AE$ ).

Los coeficientes o constantes de corte ( $K_{tc}$ ,  $K_{rc}$ ,  $K_{te}$ ,  $K_{re}$ ,  $K_{ac}$ ,  $K_{ae}$ ) utilizados en el modelado fueron obtenidos mediante un procedimiento experimental para los valores de fuerza media. Ciertamente estos coeficientes tienen un papel determinante en la precisión de la predicción del esfuerzo de corte. La capacidad de predicción de las fuerzas de corte del modelo presentado en este trabajo también dependen de coeficientes de corte obtenidos de forma experimental, y este continúa siendo un punto neurálgico en los modelos matemáticos.

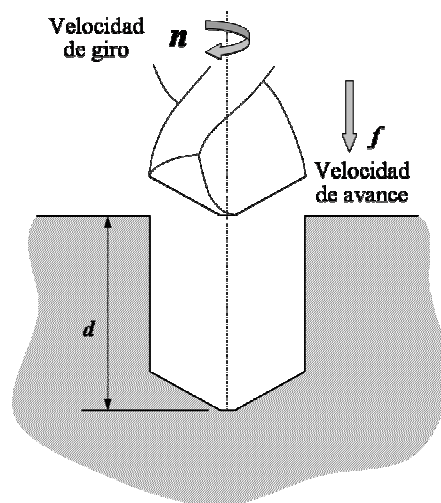
### **2.3 Diseño e implementación de un modelo dinámico computacionalmente eficiente de un proceso de fabricación complejo: el proceso de taladrado de alto rendimiento**

Otra de las técnicas de fabricación más ampliamente utilizada para crear orificios circulares en diferentes tipos de materiales es el proceso de taladrado mecánico. Tiene un amplio rango de aplicaciones desde la industria aeroespacial, automovilística y la ingeniería

civil, hasta la fabricación de dispositivos semiconductores y electrónicos. Teniendo en cuenta el total de operaciones de mecanizado en metales, el taladrado mecánico comprende aproximadamente el 40% del número total de operaciones. Sin embargo, los fondos destinados a investigación de procesos de taladrado de alto rendimiento (TAR) son mucho menores que los dedicados a otros procesos de mecanizado de alto rendimiento, tales como fresado o torneado [150]. La Ingeniería de Control y las técnicas de la Inteligencia Artificial (IA), constituyen dos piedras angulares en el ámbito de la fabricación inteligente. El empleo de sistemas multi-sensoriales y de sistemas distribuidos en red, resultan de gran utilidad para mejorar el desempeño y el rendimiento de las máquinas herramientas dotadas de control numérico por ordenador (CNC).

Los sistemas multi-sensoriales pueden proveer información sobre el estado de las operaciones de mecanizado, el estado de las herramientas de corte, posibles errores durante el mecanizado y ayudan a incrementar el rendimiento de las piezas de trabajo [151].

Los beneficios económicos de incorporar controladores distribuidos e inteligencia a la operación de taladrado de alto rendimiento, se derivan de un potencial incremento de la tasa de arranque de material a grandes velocidades de avances, del incremento del tiempo de vida útil de la herramienta y/o una reducción de la complejidad del proceso de fabricación [152].



**Figura 2.18. Esquema del proceso de taladrado de alto rendimiento.**

Durante el proceso de TAR (ver Figura 2.18), a medida que se incrementa la profundidad  $d$  en que penetra la herramienta en la pieza de trabajo, a una velocidad de avance  $f$ , se dificulta el proceso de evacuación de la viruta que se produce debido al arranque de material, lo cual conlleva un incremento de la temperatura en la punta de la herramienta, induciendo por tanto, una disminución en su vida útil y una demanda de mayor potencia de la máquina. Este fenómeno puede provocar además una mayor difusión de los elementos químicos de la

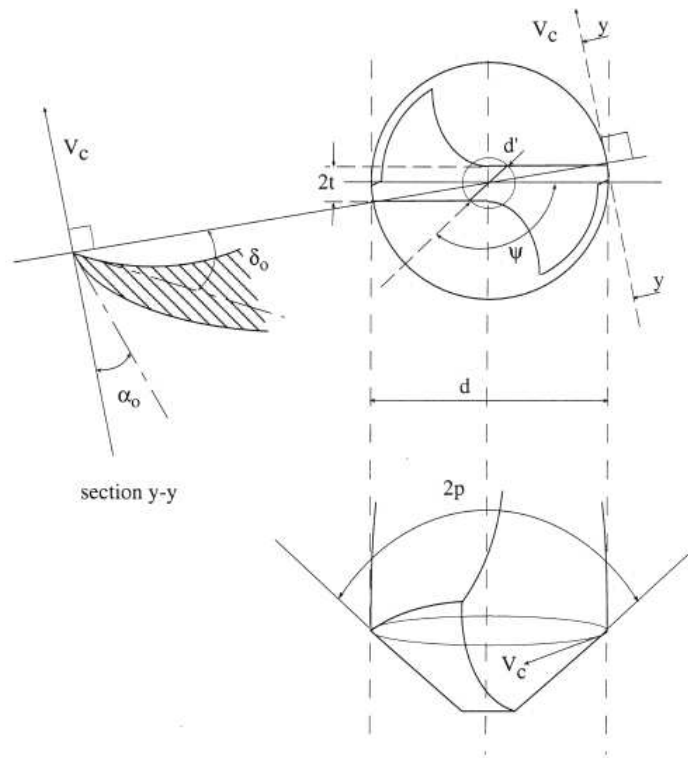
herramienta de trabajo hacia la pieza y viceversa [153]. Este fenómeno se agrava cuando se realiza una operación de taladrado de corte continuo, que no permite una evacuación de la viruta pero produce un incremento considerable en la tasa de arranque de material. Por tanto, resulta de gran utilidad el empleo de algoritmos de control sencillos y fiables, basados en técnicas de IA y capaces de trabajar en sistemas de monitorización y/o control distribuidos, para reducir el desgaste de la herramienta sin afectar la productividad en el taladrado de alto rendimiento con corte continuo.

### 2.3.1 Modelado dinámico del proceso de taladrado

Las operaciones de corte de metal, entre las cuales se encuentra el taladrado, han sido modeladas aplicando variadas técnicas. Diferentes enfoques han modelado el proceso a partir de las relaciones entre fuerza de corte y variables geométricas y de proceso, por ejemplo, velocidad de giro de la herramienta, velocidad de avance de la herramienta, profundidad de corte, etc. El diseño de sistemas de control del proceso de taladrado de alto rendimiento, basado en modelos de las fuerzas de corte, ha sido sometido a debate por la comunidad científica con anterioridad, un ejemplo de ello se muestra en un trabajo desarrollado por Landers et ál. para procesos de mecanizado [154]. Para el proceso de taladrado en particular Kim et ál. [155] analizaron, en un estudio previo, la relación entre la fuerza de empuje y la velocidad de avance en una operación de taladrado convencional realizada en una máquina herramienta con CNC. A partir del estudio se desarrolló un modelo dinámico lineal de primer orden, cuya función de transferencia es mostrada en la ecuación (2.25). En esta ecuación  $F(s)$  representa la fuerza de empuje,  $f(s)$  es la velocidad de avance de referencia y  $s$  es el operador de Laplace. Los parámetros del modelo como la ganancia ( $K$ ) y la constante de tiempo ( $T$ ), son dependientes del tipo de material que se utilice, por lo que para solventar estas incertidumbres se emplea para su ajuste una ley de control adaptativa.

$$G(s) = \frac{F(s)}{f(s)} = \frac{K}{Ts + 1} \quad (2.25)$$

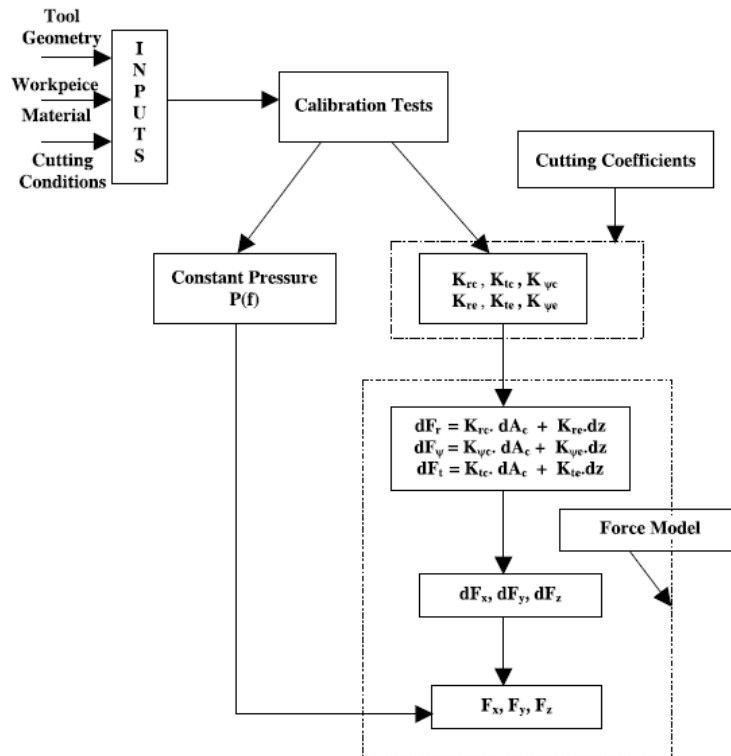
A partir del resultado anterior, Elhachimi et ál. (ver Figura 2.19) presentaron un modelo puramente geométrico de las fuerzas de empuje y de corte para brocas, validando el modelo mediante resultados experimentales de taladrado en piezas de acero [156, 157]. Recientemente se han aplicado las técnicas de IA para el modelado de la operación de taladrado, mediante el empleo de redes neuronales artificiales para predecir la fuerza existente durante el comienzo y el final de una operación de taladrado [158], como también la predicción de la fuerza en una operación de fresado en una arquitectura de control por modelo interno (IMC) basada en redes neuronales [159]. También, la máquina herramienta para operaciones de taladrado ha sido evaluada mediante modelos basados en elementos finitos, para predecir su comportamiento dinámico [160].



**Figura 2.19. Parámetros geométricos utilizados en el modelo de Elhachimi et ál.**  
(Elhachimi et ál., 1999 [156])

Por otra parte, se han realizado estudios de modelado de la operación de taladrado para el control de par de corte, con vista a reducir el desgaste de herramientas y empleando para ello modelos dinámicos generalizados de segundo orden [161]. En otro trabajo, Furness et ál. han modelado el par de corte ante cambios en los valores de referencia de la velocidad de avance de la máquina, como un sistema de tercer orden con tiempo de retraso fijo [162]. Un modelo similar de tercer orden pero sin retraso, ha sido implementado junto a un controlador proporcional, integral y derivativo (PID) para el control del par de corte en operaciones de taladrado [163].

En otro enfoque diferente para el modelado del proceso de taladrado de alto rendimiento se obtiene la respuesta dinámica a un impulso del sistema herramienta – cabezal de la máquina herramienta, mediante un golpe a la herramienta con un martillo calibrado [164] (ver Figura 2.20). Estos modelos han sido verificados, en una operación de taladrado, analizando el comportamiento de las fuerzas de corte a medida que cambia la profundidad de corte y comparándolas con la respuesta prevista por el modelo.



**Figura 2.20. Diagrama de flujo de la aproximación propuesta por Pirtini y Lazoglu.**  
(Pirtini y Lazoglu, 2005 [164])

Haciendo caso omiso por completo de los modelos basados en la caracterización física del proceso de taladrado, Oh et ál. han modelado el proceso como un sistema sub-amortiguado de segundo orden, ajustando los parámetros del modelo a partir de la respuesta temporal ante un escalón simple. Al combinar este modelo con un modelo de primer orden del comportamiento dinámico del cabezal de la máquina [163], se obtiene un modelo resultante de tercer orden:

$$G_p(s) = \frac{T(s)}{f(s)} = \frac{K_c K_s}{(\tau_{fc1}s + 1)(\tau_{fc2}s + 1)(\tau_s s + 1)} \quad (2.26)$$

donde  $T(s)$  es el par de corte,  $f(s)$  es la velocidad de avance,  $s$  es el operador de Laplace,  $K_c$  es la ganancia correspondiente al proceso de corte,  $K_s$  es la ganancia relacionada con el cabezal,  $\tau_{fc1}$  y  $\tau_{fc2}$  son las constantes de tiempo correspondientes al conjunto proceso-cabezal, y  $\tau_s$  es la constante de tiempo relacionada con el comportamiento dinámico cabezal.

Claramente existe cierta variación en el orden de los modelos disponibles en la literatura para el proceso TAR por lo que, en este trabajo, se muestra un estudio en profundidad de cual orden del sistema resulta el adecuado. Para ello, se evalúan modelos de primer, segundo y tercer orden para un sistema simple entrada-simple salida (SESS), al cual es muy fácil de aplicar las técnicas de identificación experimental.

### 2.3.2 Identificación experimental del proceso TAR en una máquina herramienta

Con vista a identificar experimentalmente el proceso de taladrado de alto rendimiento, se realizaron experimentos de taladrado, a nivel de laboratorio, en la plataforma experimental introducida en la sección 2.1.3. El carácter distribuido del sistema de adquisición instalado en la plataforma, está en concordancia con las configuraciones reales de la mayoría de los sistemas más modernos de fabricación, que sitúan los dispositivos de automatización y control en una situación remota, alejados de las unidades de mecanizado. Sin embargo, el empleo de un bus de campo como MPI o PROFIBUS, introduce retrasos temporales en la transmisión y recepción de las señales de control y variables del proceso. En el caso del bus MPI, los tiempos de transmisión y retardo son estocásticos pero están acotados, y han sido estudiados como parte de varios esquemas de control [165, 166].

Por tanto, un método sencillo y rápido para la identificación de sistemas en entornos distribuidos que tiene en cuenta las características específicas del bus de campo o de la red de datos, resulta muy importante para la aplicación de sistemas de control distribuidos a procesos de fabricación reales.

**Tabla 2.2. Resultados del análisis estadístico de los retardos de tiempo del bus de campo MPI.**

Muestras	Retardos de tiempo			
	Mínimo	Máximo	Medio	Desviación estándar
10100	9 ms	51 ms	12.58 ms	3.58 ms

Debido a que las acciones de control (señales de referencia de las velocidades de giro y de avance) son enviadas al CNC a través del bus MPI, los retardos de tiempo que este introduce deben ser incluidos en el modelo dinámico del proceso. Los retrasos de tiempo del bus MPI, en concreto el utilizado en la plataforma experimental, han sido estudiados en trabajos previos. Los resultados del estudio se muestran en la Tabla 2.2, los cuales representan solo los retrasos introducidos por el bus y no los del proceso físico en sí.

La dinámica de los componentes individuales del sistema no es analizada durante la identificación, sino que se toma el proceso completo como una sola unidad y se proporciona una función de transferencia global, que relaciona el cambio en la fuerza de corte medida  $F_{med}$  y el cambio en la velocidad de avance de referencia  $f_{ref}$ . En la Figura 2.21 se representa un diagrama de bloques teórico del sistema. El diagrama representa sendos sistemas de primer orden para el sistema de accionamientos de la máquina, el proceso de taladrado y el sistema de sensores, con retardos específicos en la dinámica de la planta,

incluidos en las funciones de transferencia de cada bloque, y en la red de comunicación. Sin embargo, es posible que fuese necesario utilizar modelos de órdenes superiores para representar el sistema completo de forma precisa. Por eso, una gran ventaja del modelo a identificar es que evalúa la dinámica del sistema global, y no cada parte individualmente, lo que facilita la incorporación de control y automatización a operaciones de mecanizado existentes.

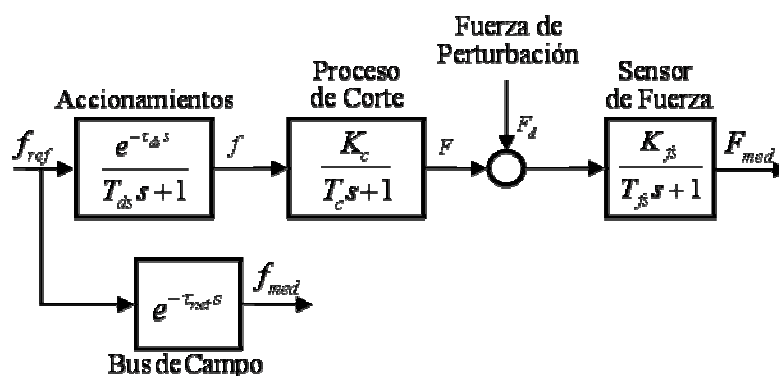


Figura 2.21. Diagrama de bloques teórico del proceso de taladrado de alto rendimiento.

Para generar y validar el modelo del proceso de TAR, se han empleado dos tipos de material: 17-4PH (un acero inoxidable endurecido por precipitación) y GGG40 (un hierro de fundición nodular dúctil). Sus propiedades se representan en la Tabla 2.3.

Tabla 2.3. Propiedades de los materiales de las piezas de trabajo experimentales.

Material	Tipo	Dureza [BHN]	Resistencia a tracción [N/mm <sup>2</sup> ]	Módulo de Young [N/mm <sup>2</sup> ]
17-4PH	Acero inoxidable	375	1310	197
GGG40	Hierro fundido	215	420	175

Ambos materiales se utilizan ampliamente en aplicaciones aeronáuticas y biomédicas y representan dos tipos muy diferentes de materiales con morfologías de viruta diferentes. Todas las herramientas de corte utilizadas son de metal duro con recubrimiento de TiN/TiAlN de 10 mm de diámetro, modelo R840-1000-30-A0A del fabricante Sandvik. Las condiciones nominales para la operación de taladrado se toman de las tablas proporcionadas por el fabricante de la herramienta.

En el procedimiento de identificación del proceso de TAR, se aplica un escalón doble en la entrada del proceso y se analiza su respuesta. Los valores de referencia o nominales de la velocidad de avance y de la velocidad de giro se programan en el CNC, junto con la trayectoria de la herramienta durante el funcionamiento. El proceso de taladrado comienza

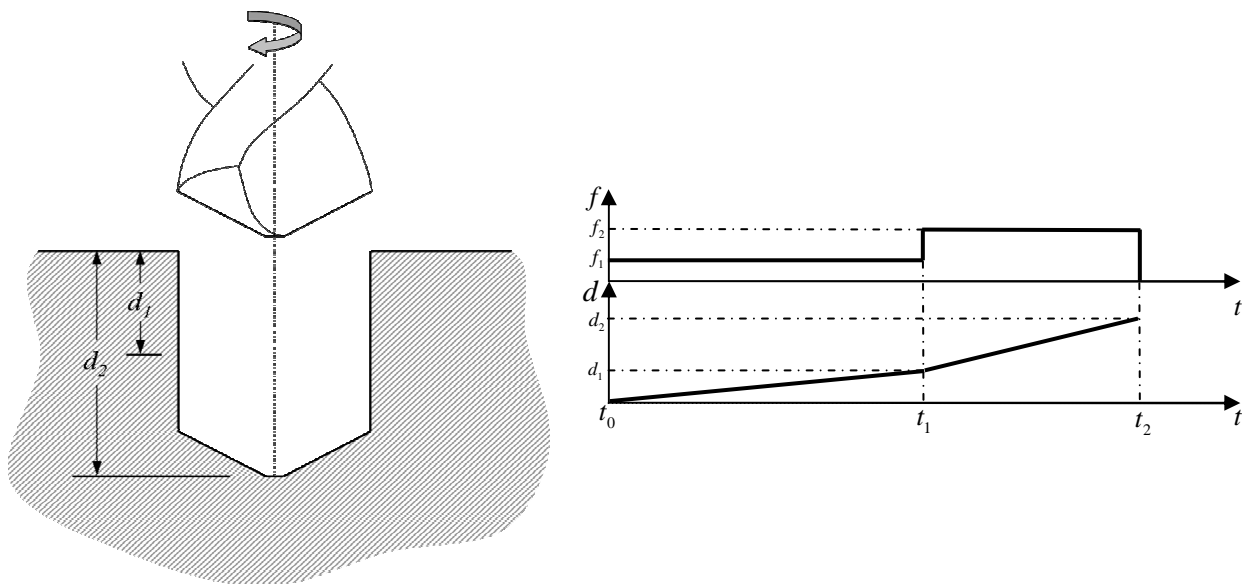


con un valor nominal en la velocidad de avance  $f_1$ , y se produce el primer escalón en la entrada cuando la herramienta contacta con la pieza en el momento  $t_0$  para comenzar el corte. Cuando la herramienta progresa hasta una profundidad  $d_1$  después de un tiempo  $t_1$ , se modifica la velocidad de avance hasta  $f_2$ , y se continúa hasta que se alcance la profundidad  $d_2$  en el momento  $t_2$ . La Tabla 2.4 muestra las condiciones nominales de taladrado para ambos materiales.

**Tabla 2.4. Condiciones nominales para la operación de taladrado.**

Material	Velocidad de giro [RPM]	Velocidad de avance [mm/min]		$d_1$ [mm]	$d_2$ [mm]
		$f_1$	$f_2$		
17-4PH	780	80	120	7	12
GGG40	870	100	120	7	12

El cambio en la velocidad de avance representa el segundo escalón en la entrada del sistema, que incluye retardos en el proceso de taladrado y en las señales de referencia enviadas al CNC. La Figura 2.22 muestra una representación gráfica del perfil del taladro utilizado en los experimentos.



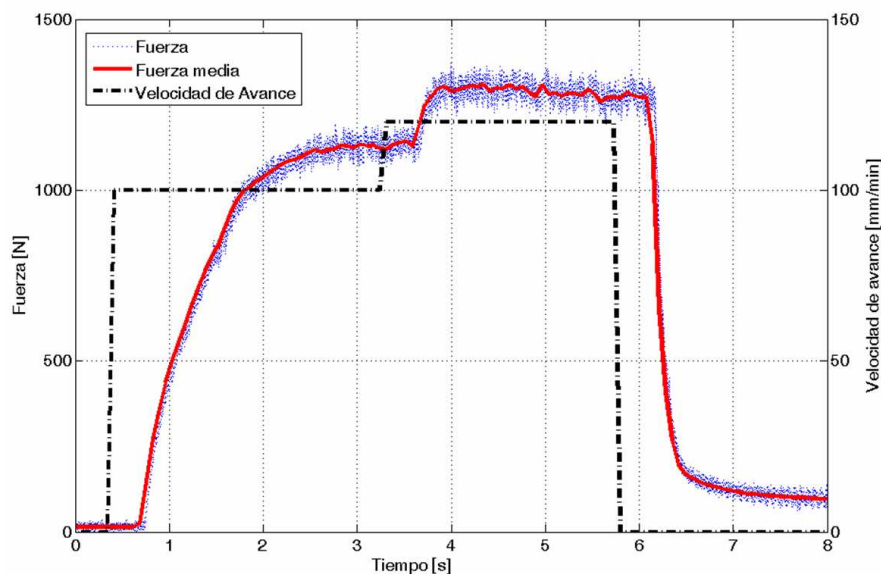
**Figura 2.22. Perfil para la operación de taladrado.**

Los datos de fuerza se filtran mediante un filtro FIR utilizando el algoritmo de media móvil y evitando introducir retrasos temporales durante el procesamiento (ver sección 2.1.1). La longitud de la ventana de filtrado  $ws$  es igual al número de muestras obtenidas durante

una revolución de la herramienta (2.27); por tanto, el tamaño de la ventana depende de la velocidad de giro del taladro  $n$  y de la frecuencia de adquisición de datos  $SF$ .

$$ws = SF \cdot 60 / n \quad (2.27)$$

Para cada material, con las mismas condiciones nominales, se realizaron tres réplicas del experimento, de tal manera que el modelo obtenido a partir de un conjunto de datos pueda validarse con datos diferentes.



**Figura 2.23.** Señal de entrada al proceso (velocidad de avance) y fuerza obtenida, bruta y media (filtrada) de una operación de taladrado en el material GGG40.

**Tabla 2.5.** Modelos del taladrado de GGG40.

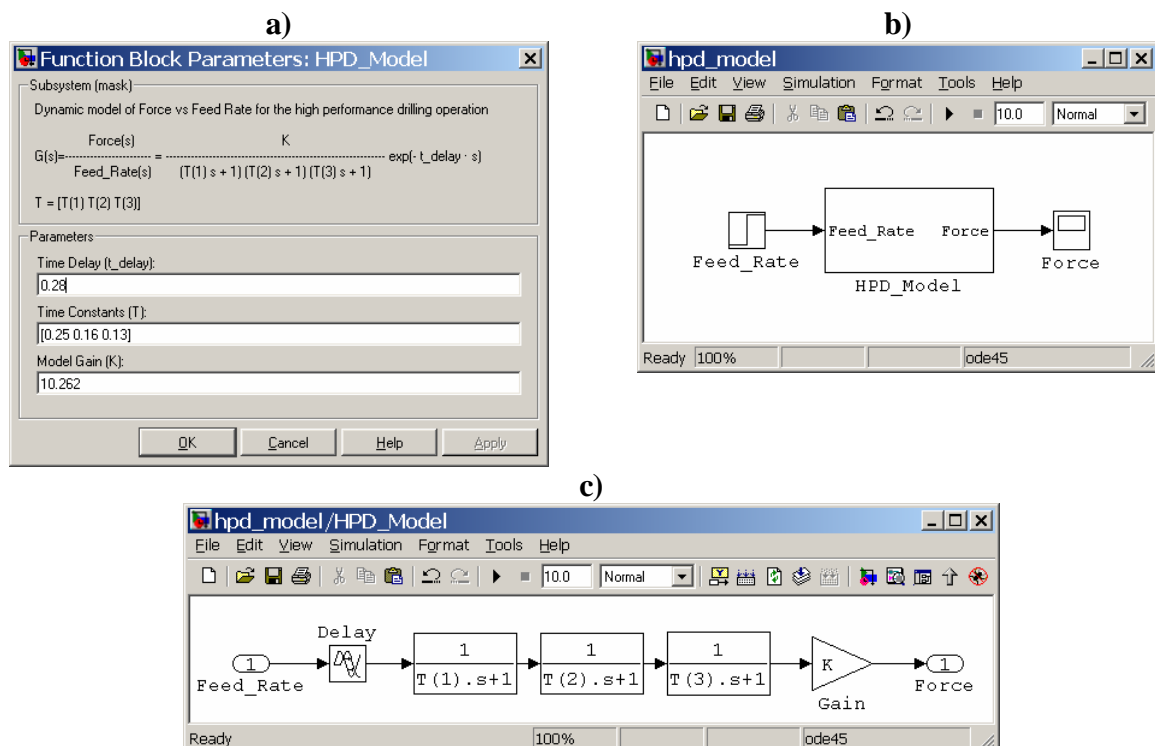
Modelo	Función de Transferencia
1er orden	$G(s) = \frac{10.272e^{-0.28s}}{0.38s+1}$
2do orden	$G(s) = \frac{10.232e^{-0.28s}}{(0.25s+1)(0.16s+1)}$
3er orden	$G(s) = \frac{10.262e^{-0.28s}}{(0.25s+1)(0.16s+1)(0.13s+1)}$

La Figura 2.23 muestra las señales de fuerza filtradas y sin filtrar, junto con la velocidad de avance de una operación de taladro en el material GGG40. Claramente, el filtrado reduce el ruido de alta frecuencia presente en la señal de fuerza, lo que ayuda a que la creación del modelo sea más fácil sin pérdida resolución de datos temporales. Además, el retardo en la salida debido tanto a la arquitectura del medio de conexión como a la propia planta física es claramente visible en esta figura.

A partir de la aplicación de las técnicas de minimización del error cuadrático a los datos obtenidos, se ajustaron modelos lineales de primer, segundo y tercer orden. En la Tabla 2.5 se relacionan las funciones de transferencia de tiempo continuo que describen los modelos obtenidos. Las componentes dinámicas difieren, aunque la única diferencia entre los modelos de segundo y tercer orden es la adición de un polo más y una pequeña modificación en la ganancia. Todos ellos tienen el mismo tiempo de retardo. El uso de una función de dos escalones en la velocidad de avance para la obtención de los datos permite obtener un modelo más realista que incorpora una variedad de situaciones de taladrado.

### 2.3.3 Implementación de los modelos en Matlab/Simulink

Con vistas a simular los modelos ajustados mediante algoritmos computacionalmente eficientes, para su posterior análisis o validación, o para su empleo futuro en diseño y ajuste de algoritmos de control, se ha realizado su implementación en Matlab/Simulink. La aplicación desarrollada en Simulink permite la simulación de cualquiera de los tres modelos ajustados.

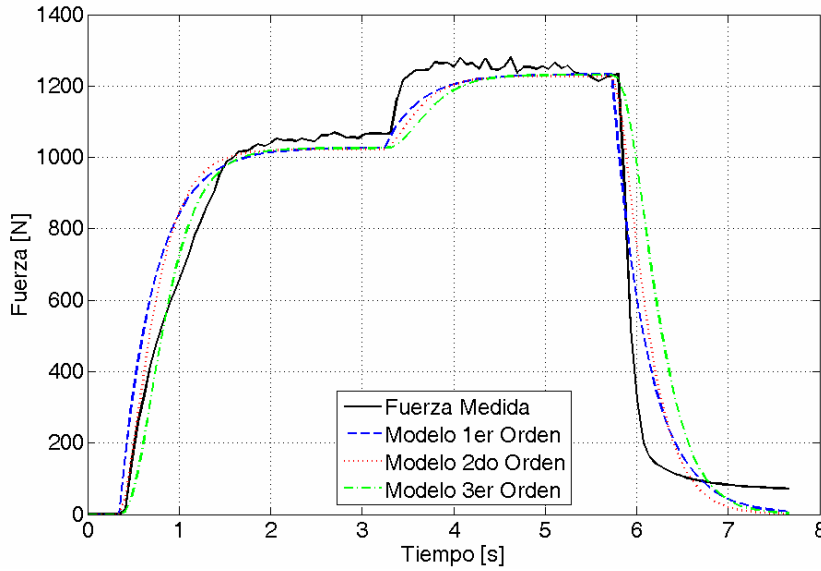


**Figura 2.24. Implementación de los modelos en Matlab/Simulink.**

a) Entrada de parámetros al modelo, b) entrada y salida de datos del modelo, c) diagrama de bloques de la función de transferencia del modelo.

El algoritmo consta principalmente de un diagrama de bloques que implementan todos los componentes de la función de transferencia del modelo (Figura 2.24.c) y de una interfaz de usuario para introducir sus parámetros (Figura 2.24.a). Tanto los polos del modelo, como el

retardo de transporte, son implementados con los bloques disponibles en la librería de sistemas continuos. Además, se ha utilizado como método numérico de resolución de ecuaciones diferenciales ordinarias, el algoritmo de Dorman-Price (Runge-Kutta (4,5)) [167].



**Figura 2.25.** Respuesta de los modelos de primer, segundo y tercer orden de la fuerza ante cambios en la velocidad de avance y conjunto de datos de validación.

Los tres modelos han sido simulados ante el mismo cambio de doble escalón de la velocidad de avance (entrada *Feed\_Rate* de la aplicación) empleado durante la etapa experimental. En la Figura 2.25 se representa la respuesta de los tres modelos junto a la fuerza filtrada de otro conjunto de datos de validación.

### 2.3.4 Validación de los modelos y análisis de error de ajuste

La precisión de los modelos es evaluada a partir del cálculo de dos índices de comportamientos basados en error: la integral del error absoluto (*IAE*), ecuación (2.28), y la integral del error cuadrático (*ISE*), ecuación (2.29). Estos dos criterios de error son ampliamente utilizados para cuantificar el grado de exactitud de ajuste de un modelo determinado.

$$IAE = \sum_{i=1}^n |y(i) - \hat{y}(i)| / \sum_{i=1}^n y(i) \quad (2.28)$$

$$ISE = \sqrt{\sum_{i=1}^n (y(i) - \hat{y}(i))^2 / \sum_{i=1}^n y^2(i)} \quad (2.29)$$

Los índices de comportamiento son calculados a partir del error,  $e(t)$ , entre la salida del modelo,  $\hat{y}(t)$ , y los datos de validación,  $y(t)$ , tal como se expresa en la ecuación (2.6). En la Tabla 2.6 se relacionan los índices de error calculados para cada modelo. En ambos índices, el menor error corresponde al modelo de primer orden, aunque el modelo de segundo orden se comporta de modo similar. El modelo de tercer orden es el que peor se comporta en este análisis, a pesar de que proporciona una representación mejor del escalón inicial.

**Tabla 2.6. Resultados del análisis de error para los modelos del taladrado de GGG40.**

Modelo	IAE %	ISE %
1er orden	8.31	10.13
2do orden	8.59	11.24
3er orden	10.12	16.07

Por otra parte, la validación de la componente dinámica de los modelos, se realiza mediante un análisis de auto-correlación de los residuos del error, empleando para ello una función de auto-correlación descrita en la ecuación (2.30). Empleando esta función, el error del modelo se compara muestra a muestra para determinar si existe alguna contribución residual de los datos, es decir, si hay componentes dinámicas que no están bien representadas por el modelo.

$$\hat{p}_k = \frac{\sum_{i=k}^{N-(k+1)} (y(i) - \hat{y}(i))(y(i+k) - \hat{y}(i))}{\sum_{i=0}^{N-1} (y(i) - \hat{y}(i))^2} \quad (2.30)$$

La Figura 2.26 muestra el análisis de correlación, de los tres modelos del taladrado del GGG40, realizado a partir de los datos de validación. Los tres modelos se ajustan al intervalo de confianza del 95%, por lo que todos pueden representar la dinámica del sistema de forma adecuada. Si embargo, los modelos de segundo y tercer orden representan la dinámica del sistema un poco mejor que el modelo de primer orden, lo cual es un resultado opuesto al procedimiento de análisis de error. El resultado anterior es de esperarse, debido a que un modelo de primer orden ignora dinámica de órdenes superiores. Por otra parte, aunque estos modelos puedan realizar un buen ajuste de los datos y provean una adecuada representación de la dinámica del proceso, puede que no sean apropiados para ser utilizados en esquemas de control a lazo cerrado debido a su potencial inestabilidad. Por lo que resulta necesario realizar simulaciones para determinar la aplicabilidad del modelo en cualquier esquema de control.

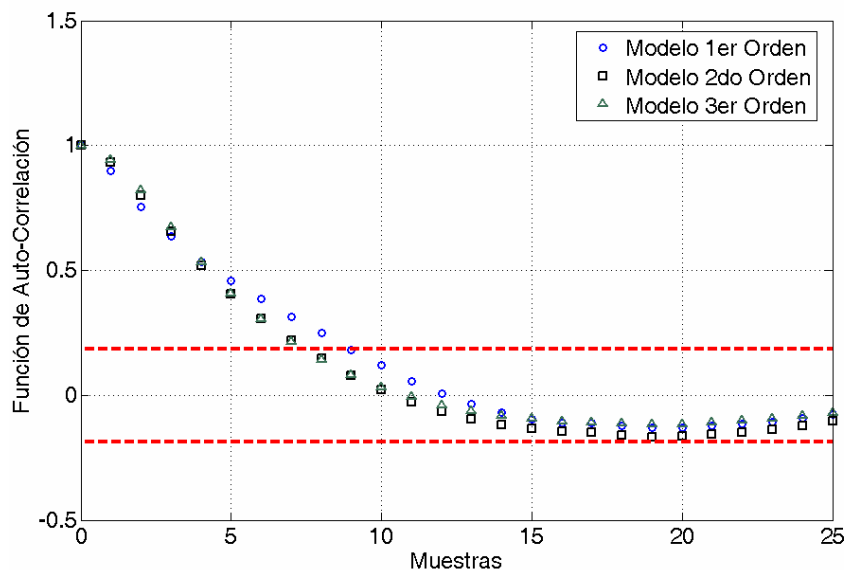


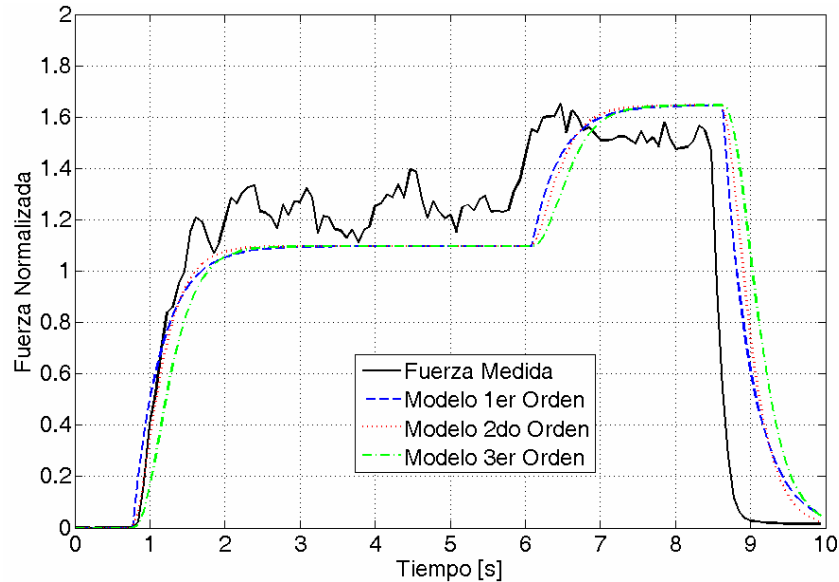
Figura 2.26. Función de auto-correlación del error en la predicción de fuerza en el taladrado de GGG40.

### 2.3.5 Aplicación de los modelos a la predicción de las fuerzas de corte en materiales de mayor dureza. Análisis de errores

Como se ha mencionado previamente, la aplicación de un modelo lineal de un proceso de taladrado creado a partir del taladrado de un material a otro material, es un problema actual en sistemas de producción. Los modelos identificados para el mecanizado del material GGG40, serán empleados para la predicción de la fuerza en el taladrado de un material de mayor dureza como acero inoxidable 17-4PH. Para ello primeramente se debe reescalar la ganancia de las funciones de transferencia a la unidad. Para reescalar los datos experimentales, los valores de fuerza y velocidad de avance se dividen entre sus valores medios durante todo el intervalo de análisis. De este modo, se realiza una comparación directa del comportamiento dinámico del proceso de TAR para cada material. La Figura 2.27 muestra la aplicación de los modelos del GGG40 con ganancia unidad a un conjunto de datos normalizados del material 17-4PH. En general, se predicen bastante bien las salidas de ambos escalones, lo que indica que la dinámica del proceso de taladrado se mantiene similar ante cambios de material.

Se observa una diferencia significativa en el instante de tiempo en que comienza el segundo escalón entre los datos de validación y la predicción del modelo. Esto podría deberse a un retardo del medio de conexión (5) no constante, que se ha modelado como un retardo fijo, o debido al cambio de punto de operación entre materiales diferentes. En este caso, el retardo modelado es más largo que el retardo real, constituyendo un factor de seguridad al aplicar una estrategia de control. También aparece una no linealidad en la ganancia estática, como se observa de la subestimación de la fuerza después del primer escalón y la sobreestimación de la misma después del segundo. Esto se corresponde bien con la naturaleza

no lineal de modelos geométricos de la fuerza de corte estática durante el taladrado, presentados con anterioridad en la literatura por Elhachimi [156], Hamade [150] y otros.



**Figura 2.27.** Aplicación de los modelos del GGG40 con ganancia unidad a la fuerza normalizada del taladrado del material 17-4PH.

La Tabla 2.7 muestra los índices de error que se obtienen al aplicar los modelos de taladrado obtenido con los datos del material GGG40 al material 17-4PH. Los índices de comportamiento muestran que el los modelos constituyen una representación adecuada del proceso de taladrado de alto rendimiento del acero inoxidable 17-4PH. Nuevamente, el modelo que realiza un mejor ajuste es el de primer orden, seguido del segundo.

**Tabla 2.7.** Análisis del error de ajuste para los modelos generados del material GGG40 empleados en el material 17-4PH.

Modelo	<i>IAE</i> %	<i>ISE</i> %
1er orden	18.20	24.23
2do orden	19.28	26.87
3er orden	23.18	31.69

## 2.4 Diseño e implementación de un modelo dinámico múltiple entrada-simple salida, computacionalmente eficiente de un proceso de fabricación complejo. Estudio del caso particular del proceso de fresado de alta velocidad

El estudio, desarrollo e implementación del modelo físico del proceso de fresado de alta velocidad (FAV), ha permitido obtener un mayor conocimiento de este y a partir del cual se ha podido determinar algunas de las variables y parámetros que tienen una mayor influencia en cambios en las fuerzas de corte. Por lo que sería interesante caracterizar dinámicamente la influencia de estas variables y parámetros en el proceso, teniendo en cuenta la experiencia obtenida de aplicar las técnicas de identificación a un proceso de fabricación complejo simple entrada-simple salida, tal como el proceso de taladrado de alto rendimiento.

El uso del valor instantáneo de las fuerzas de corte como señal, no es de utilidad para la obtención de modelos lineales e invariantes en el tiempo, mediante las técnicas de identificación estudiadas. Sin embargo, la caracterización de las fuerzas a través de su valor eficaz, medio cuadrático o simplemente su valor medio, puede ser ventajoso para caracterizar dinámicamente el proceso, a partir de cambios en algunas de sus variables más significativas. Un modelo dinámico del proceso de FAV resulta provechoso para el diseño de algoritmos de control inteligente que conlleven una mejor optimización del proceso, ya sea para un potencial incremento de la tasa de arranque de material, incrementar el tiempo de vida útil de la herramienta, entre otros beneficios.

Utilizando como variable del proceso la fuerza resultante  $F_{total}(\phi)$ , la cual se define según la ecuación (2.23) del modelo físico del FAV, y que no es más que la norma *euclídea* del vector de fuerzas de tres componentes  $[F_x; F_y; F_z]$ , la fuerza media en cada revolución de la herramienta se obtiene como:

$$F_m = \frac{1}{2\pi} \int_0^{2\pi} F_{total} d\phi \quad (2.31)$$

Empleando como variable de salida del proceso de FAV, la fuerza resultante media  $F_m$ , este se puede representar como un proceso dinámico múltiple entrada-simple salida (MISO), cuyas variables de entrada serían la velocidad de avance de referencia de la herramienta con respecto a la pieza  $f$ , la velocidad de giro de referencia de la herramienta  $n$  y la profundidad axial de corte relativa entre la herramienta y la superficie de la pieza  $ap$ . Algunos de los parámetros que mayor influencia tienen en el comportamiento de las fuerzas son el diámetro de la herramienta  $D$ , la profundidad radial de corte  $ae$ , la dureza del material, entre otros. La Figura 2.28 ilustra una representación gráfica del proceso, por otra parte, si se considera el proceso como un sistema lineal e invariante en el tiempo, en cierto intervalo de tiempo, este



puede modelarse mediante funciones de transferencia en el operador de Laplace, tal como muestra también el diagrama de bloques ilustrado en la figura. En el modelo  $F_m(s)$  representa los cambios en la fuerza de corte resultante media,  $f(s)$  cambios en la velocidad de avance,  $n(s)$  cambios en la velocidad de giro y  $ap(s)$  representa cambios en la profundidad axial de corte relativa entre la herramienta y la superficie de la pieza. Esta última variable puede ser considerada también como una variable de perturbación del proceso.

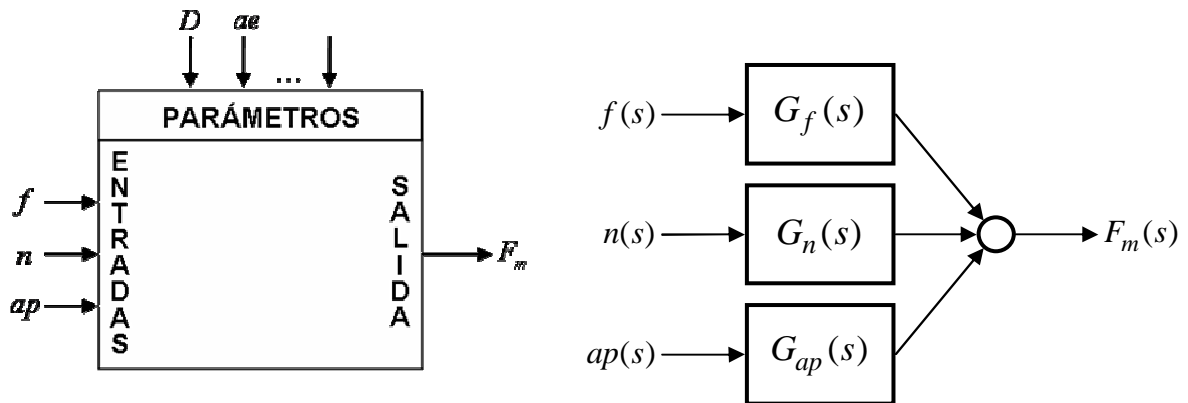


Figura 2.28. Esquema del proceso de FAV y diagrama de bloques de una representación del proceso mediante funciones de transferencia.

#### 2.4.1 Diseño de experimento para la identificación del modelo dinámico del FAV

Con el objetivo de identificar experimentalmente el proceso se utilizará como probeta una pieza de acero de alta dureza 17-4PH, con un perfil escalonado en su superficie que presenta cambios en escalón de 0.5 mm (ver Figura 2.29). El perfil escalonado permitiría estimar el comportamiento de la fuerza media de corte ante cambios de tipo escalón en la profundidad axial  $ap$ . Como puede observarse en la figura, el perfil está compuesto por dos escalones de subida y dos de bajada, es decir, incrementos y decrementos de  $ap$ , tomándose como valor inicial de la profundidad axial igual también a 0.5 mm ( $ap_0$ ).

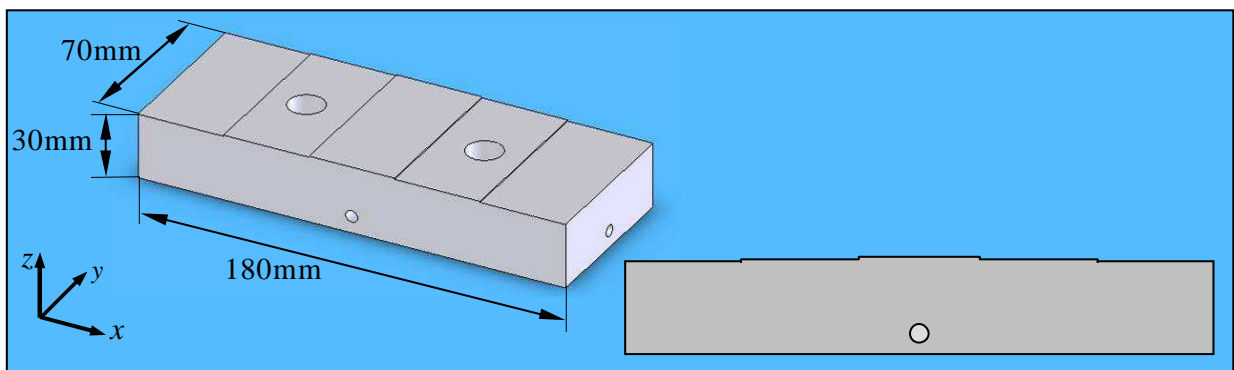
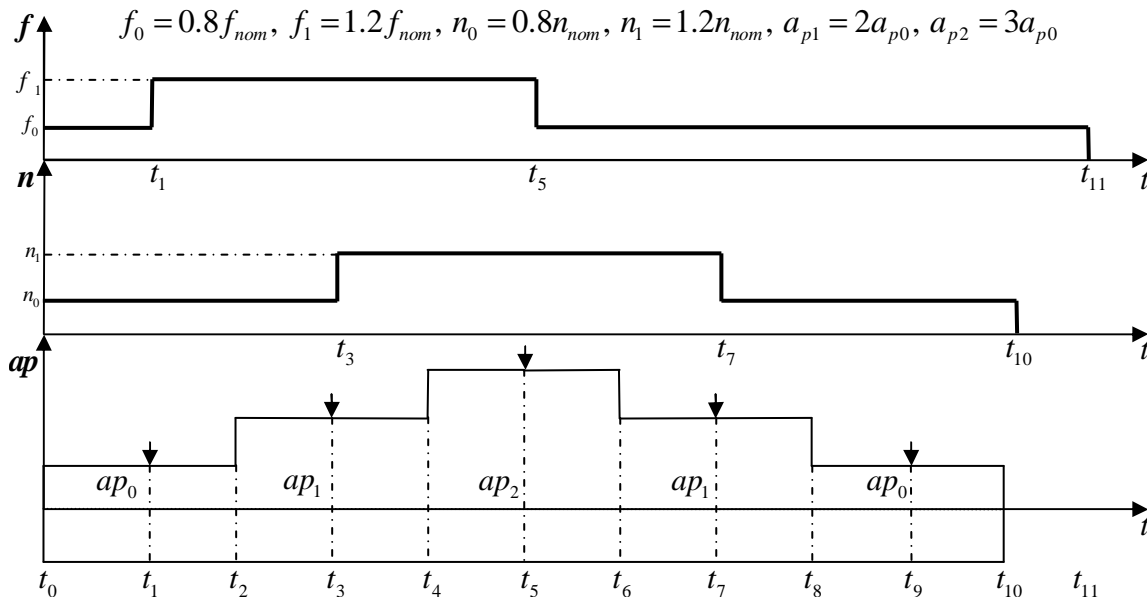


Figura 2.29. Probeta con perfil escalonado en su superficie con cambios de 0.5 mm.

Para el procedimiento de identificación, se realizará una operación de ranurado a lo largo del eje  $x$  de la pieza. Para esta operación la profundidad radial de corte es igual al diámetro de la herramienta, es decir  $ae = D$ . De la misma forma, durante la operación de corte, para identificar la influencia de los cambios en las otras dos variables de entrada al modelo, velocidad de avance  $f$  y velocidad de giro, se forzarían, para ambas variables, dos cambios de tipo escalón, también uno positivo y otro negativo. El valor de los cambios estará relacionado con los valores nominales recomendados en manuales de mecanizado para este tipo de material. La operación de corte se realizaría de tal manera que los cambios en cada uno de las variables no se solapen, de tal forma que el cambio que experimente la fuerza de corte media, sea debido a una sola variable de entrada al proceso.



**Figura 2.30. Diseño de experimento para la identificación del proceso de FAV. Cambios planificados para cada una de las variables de entrada al proceso.**

En la Figura 2.30 se representa gráficamente y en detalle el diseño del experimento, con los instantes de tiempo en que se produciría cada uno de los cambios en las variables de entrada. La Tabla 2.8 relaciona los valores que alcanzarían cada una de las variables de entrada, durante la operación de corte, a lo largo del eje  $x$  de la pieza.

Se realizarán dos tipos de experimentos, con herramientas de dos diámetros diferentes de 10 y 12 mm, de tal forma que el modelo que se obtenga contenga la influencia de este parámetro. En la Tabla 2.9 se describen las condiciones nominales de corte para cada uno de los experimentos.

**Tabla 2.8.** Valores que alcanzan cada una de las variables de entrada a lo largo del eje  $x$  de la pieza, durante la operación de ranurado.

$t$ [seg]	$x$ [mm]	$f$ [mm/min]	$n$ [rpm]	$ap$ [mm]
$0$	$0$	$f_0$	$n_0$	$ap_0$
$t_1$	$18$	$f_1$	$n_0$	$ap_0$
$t_2$	$36$	$f_1$	$n_0$	$ap_1$
$t_3$	$54$	$f_1$	$n_1$	$ap_1$
$t_4$	$72$	$f_1$	$n_1$	$ap_2$
$t_5$	$90$	$f_1$	$n_1$	$ap_2$
$t_6$	$108$	$f_0$	$n_1$	$ap_1$
$t_7$	$126$	$f_0$	$n_0$	$ap_1$
$t_8$	$144$	$f_0$	$n_0$	$ap_0$
$t_9$	$162$	$f_0$	$n_0$	$ap_0$
$t_{10}$	$180$	$f_0$	$0$	$0$
$t_{11}$	$198$	$0$	$0$	$0$

**Tabla 2.9.** Condiciones nominales para el experimento de identificación.

Diám. de la herram. [mm]	Veloc. de giro $n_{nom}$ [rpm]	Velocidad de avance $f_{nom}$ [mm/min]	$ae$ [mm]	$ap$ MAX [mm]	Ref. Herramienta
10	1432	215	10	1.5	R216.34-10050-AK22H 1620 Sandvik
12	1193	298	12	1.5	R216.34-12050-AK26H 1620 Sandvik

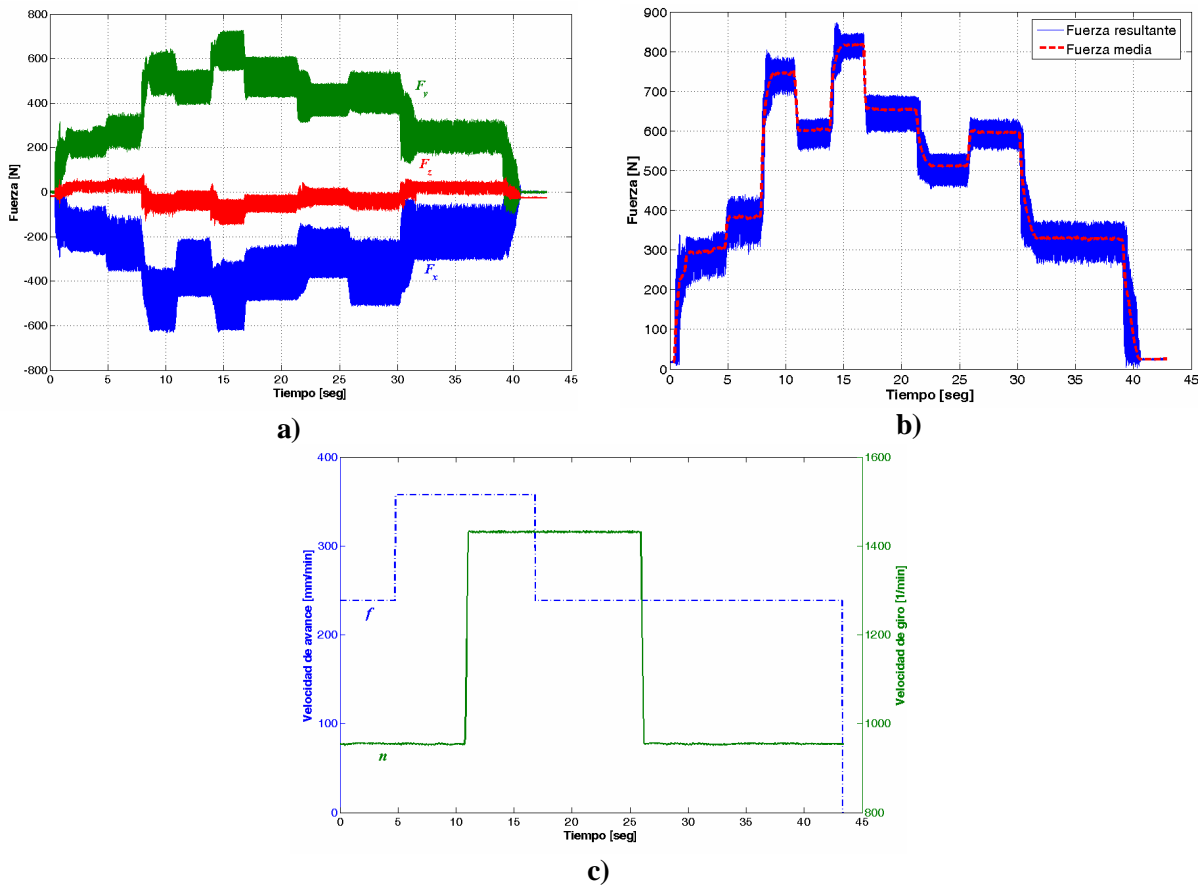
### 2.4.2 Identificación experimental del modelo dinámico del proceso de fresado de alta velocidad

Se realizaron ensayos experimentales en la plataforma descrita en la sección 2.1.2, ejecutándose 3 réplicas del ensayo para la herramienta de 10 mm y 4 para la herramienta de 12 mm. Antes de realizar el procedimiento de identificación se pre-procesaron los datos registrados. De las tres componentes cartesianas de la fuerza se obtuvo la fuerza resultante, a la cual se le aplicó el filtro FIR con el algoritmo de media móvil y de tal forma que no se introdujese retardo de tiempo en la señal filtrada durante el pre-procesamiento de los datos.

La longitud de la ventana de filtrado es igual al número de muestras obtenidas durante una vuelta de la herramienta. Debido a que la velocidad de giro cambia durante cada experimento, se determinó cada uno de los instantes de tiempo en que se producían los cambios en la velocidad, quedando dividido el vector de fuerza resultante en tres intervalos de tiempo, el primero se corresponde a  $n_0 = 0.8n_{nom}$ , el segundo a  $n_1 = 1.2n_{nom}$  y el tercero nuevamente a

$n_0 = 0.8n_{nom}$ . En cada uno de los intervalos cambia el tamaño de la ventana de datos para el filtrado.

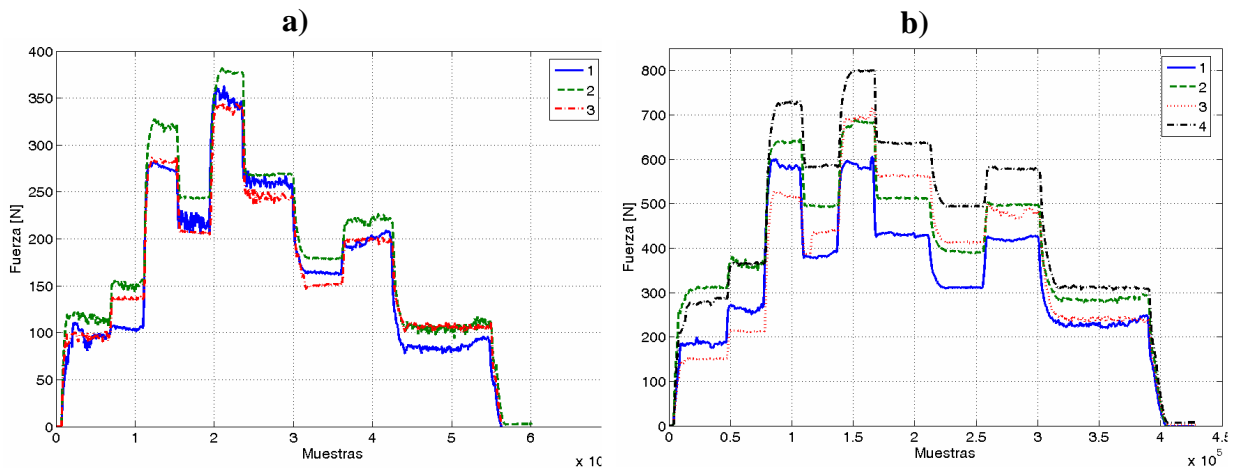
En la Figura 2.31 se muestra el comportamiento de las tres componentes de fuerzas ( $F_x; F_y; F_z$ ), de la fuerza resultante y de la fuerza media, en una de las réplicas de la operación de corte con la herramienta de 12 mm. En la misma figura, se ilustra además, los cambios en las referencias de las velocidades de giro y de avance, durante todo el corte.



**Figura 2.31. Señales registradas durante la operación de fresado con la herramienta de 12 mm.**  
a) Fuerzas en los tres ejes, b) fuerza resultante y media y c) velocidades de giro y avance.

Tras el filtrado de los datos de fuerza en cada uno de los experimentos, se analizó el comportamiento de la fuerza media en cada uno de ellos, para decidir de cuáles datos se podría obtener un mejor ajuste. La Figura 2.32 muestra la fuerza media de cada una de las réplicas de los experimentos para las herramientas de 10 y 12 mm respectivamente. Del experimento con la herramienta de 10 mm, se escogieron la segunda y tercera réplica para la obtención de los modelos con esta herramienta, debido a que en la primera réplica no se aprecia claramente el cambio en la fuerza producto del primer escalón en la velocidad de avance. De la misma forma, del experimento con la herramienta de 12 mm, se escogieron la

primera y la segunda réplica para obtener el modelo correspondiente, no utilizando el tercer experimento porque se aprecia un cambio inesperado en la fuerza debido al primer cambio en la velocidad de giro, dejando la cuarta réplica del experimento para la validación de los modelos obtenidos.



**Figura 2.32. Fuerza media en cada uno de los experimentos.**  
a) Herramienta de 10 mm y b) herramienta de 12 mm.

### *Obtención del modelo*

El modelo dinámico del proceso de FAV será obtenido mediante mínimos cuadrados y aplicando el procedimiento descrito para sistemas simple entrada-simple salida. Para ello se identificarán las funciones de transferencia asociada a cada variable de entrada al modelo, considerando cada una de estas funciones componentes del modelo, como un sub-sistema de simple entrada-simple salida, siendo necesario entonces determinar el instante de tiempo en que ocurre el cambio en escalón de su variable de entrada correspondiente. Con respecto a las velocidades de giro y de avance se determinaron los cambios a partir de las propias señales medidas. Por otra parte, los instantes de tiempo donde ocurren los cambios en la profundidad axial de corte, fueron establecidos mediante el análisis de la señal de fuerza media.

Con el objetivo de reconstruir la señal correspondiente a los cambios en la profundidad axial de corte, se midió en una de las probetas donde se realizaron los ensayos, la profundidad real a la que se realizaron los experimentos. Para ello se empleó un instrumento de tipo palpador con una precisión de  $1\mu\text{m}$ . A través de este instrumento se comprobó que la profundidad de corte inicial es de 0.42 mm exactamente y de 0.5 mm para el resto de los cambios.

Determinados los instantes de tiempo correspondientes a cada cambio en las variables de entrada y teniendo los vectores de datos de las variables de entrada y de salida, se distribuyeron los datos para cada instante de tiempo. Se obtuvieron 9 nuevos subconjuntos de

datos para cada ensayo, definiéndose en el siguiente orden y nombrados según el cambio de la variable a que corresponden:  $0 - ap_0$ ;  $f_0 - f_1$ ;  $ap_0 - ap_1$ ;  $s_0 - s_1$ ;  $ap_1 - ap_2$ ;  $f_1 - f_0$ ;  $ap_2 - ap_1$ ;  $s_1 - s_0$ ;  $ap_1 - ap_0$ .

**Tabla 2.10. Parámetros de algunos de los modelos obtenidos con la herramienta de 12 mm.**

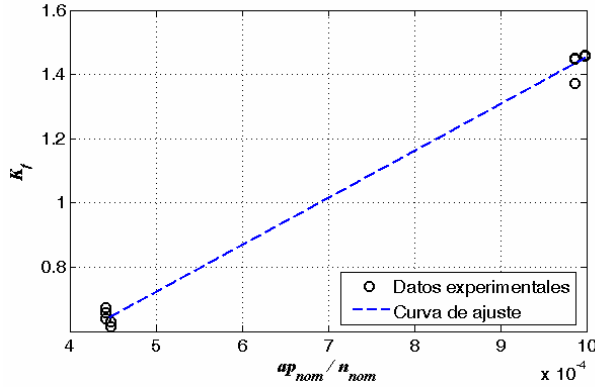
Rép.	ap	f	n	f/n	TM	Subconj.	Aj. %	K	Tp1	Tp2	Td
1	0.422	238.4	954.4	0.2498	P2	$0 - ap_0$	86.76	434.55	0.1776	0.178	0
1	0.422	357.6	954.4	0.3747	P2D	$f_0 - f_1$	85.17	0.639	0.1817	0.117	0.0037
2	0.938	357.6	954.4	0.3747	P2	$ap_0 - ap_1$	97.8	504.77	0.1677	0.0520	0
2	0.938	357.6	1431.6	0.2498	P2D	$s_0 - s_1$	93.439	-0.302	0.0351	0.0437	0.0262
1	1.413	357.6	1431.6	0.2498	P1	$ap_1 - ap_2$	88.01	414.02	0.161	0	0
1	1.413	238.4	1431.6	0.1665	P1D	$f_1 - f_0$	84.97	1.449	0.0809	0	0.03
2	0.929	238.4	1431.6	0.1665	P2D	$ap_2 - ap_1$	96.417	232.25	0.385	0.0948	0.0299
1	0.917	238.4	954.4	0.2498	P2	$s_1 - s_0$	91.06	-0.225	0.0152	0.0162	0
2	0.415	238.4	954.4	0.2498	P2	$ap_1 - ap_0$	93.235	392.91	0.512	0.115	0

A partir de la señal de fuerza particionada en cada subconjunto, se identificaron para cada uno de ellos, tanto funciones de primer (tipo de modelo P1) como de segundo (tipo de modelo P2) orden, con raíces reales, sin retraso y con retraso de tiempo (P1D o P2D). En la Tabla 2.10 se muestra un extracto de los parámetros de los modelos obtenidos con la herramienta de 12 mm, siendo: la primera columna el número de réplica del experimento; de la segunda a la cuarta los valores que se alcanzan de la profundidad axial de corte, la velocidad de avance y la velocidad de giro respectivamente; la quinta columna constituye el avance por revolución de la herramienta, que se expresa en mm/rev; la sexta columna representa el tipo de modelo, en la séptima columna se muestra el porcentaje de ajuste del modelo calculado como el error cuadrático medio entre el modelo ajustado y los datos experimentales y las últimas 4 columnas constituyen los parámetros de los modelos, siendo **K** la ganancia del modelo, **Tp1** la primera constante de tiempo, **Tp2** la segunda constante de tiempo y **Td** el tiempo de retraso.

Una vez obtenidos los diferentes tipos de modelos de los sub-sistemas para cada cambio de las variables de entrada, se buscarán relaciones matemáticas, mediante la aplicación de técnicas de regresión, entre los parámetros de los modelos y los valores nominales del resto de las variables de entrada al modelo MISO que no intervienen en ese sub-sistema, en el instante que se produce el cambio. A modo de ejemplo, en la ecuación (2.32) se describe la función de transferencia del sub-sistema que relaciona el cambio en la fuerza media de corte y el cambio en la velocidad de avance, se buscará alguna relación que permita determinar la ganancia del modelo  $K_f$  a partir de los valores nominales de la velocidad de giro  $n_{nom}$  y de la profundidad axial de corte  $ap_{nom}$ , durante el cambio en la velocidad de avance. De la misma forma, se buscarán las relaciones matemáticas para el resto de sub-sistemas.

$$G_f(s) = \frac{F_m(s)}{f(s)} \Rightarrow K_f = g(n_{nom}, ap_{nom}) \quad (2.32)$$

a)

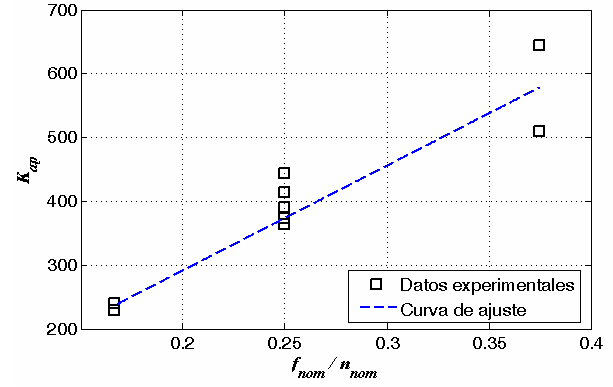


$$f(x) = p_1 x + p_2$$

$$p_1 = 1463 \quad [1408, 1518]$$

$$p_2 = -0.008977 \quad [-0.05099, 0.03304]$$

b)

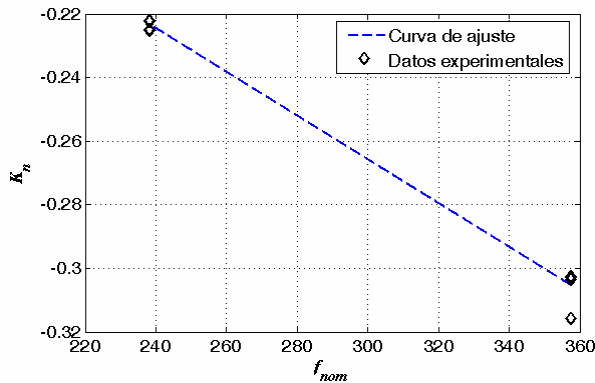


$$f(x) = p_1 x + p_2$$

$$p_1 = 1640 \quad [1100, 2180]$$

$$p_2 = -36.19 \quad [-183.8, 111.5]$$

c)



$$f(x) = p_1 x + p_2$$

$$p_1 = -0.0006879 \quad [-0.0007297, -0.0006461]$$

$$p_2 = -0.05945 \quad [-0.07175, -0.04715]$$

**Figura 2.33. Funciones de ajuste para las ganancias de los sub-sistemas que componen el modelo.**  
Caso herramienta de 12 mm: a)  $K_f$  vs  $ap_{nom} / n_{nom}$ ; b)  $K_{ap}$  vs  $f_{nom} / n_{nom}$  y c)  $K_n$  vs  $f_{nom}$ .

La Figura 2.33 representa la relación de las ganancias entre los cambios en la fuerza media y las variables de entrada de cada sub-sistema, para los ensayos con la herramienta de 12 mm. La Figura 2.33.a describe la curva lineal de ajuste de la relación entre ganancia  $K_f$  del sub-sistema  $G_f(s)$  y el coeficiente entre la profundidad de corte nominal y la velocidad de giro nominal durante el instante de cambio de la velocidad de avance, es decir,  $K_f$  vs  $ap_{nom} / n_{nom}$ . De la misma forma, en la Figura 2.33.b se ilustra el ajuste de la función

que relaciona la ganancia  $K_{ap}$  con el avance por revolución nominal en el instante de cambio de la profundidad axial de corte, es decir,  $K_{ap}$  vs  $f_{nom}/n_{nom}$ . También en esa misma figura se muestra la función ajustada que representa la relación  $K_n$  vs  $f_{nom}$ , es decir, solamente se ha tenido en cuenta el valor nominal de la velocidad de avance. Para cada función de ajuste se relacionan sus parámetros y sus valores para un 95% del límite de confianza.

De la misma forma se determinaron la relación entre las ganancias de los sub-sistemas para la herramienta de 10 mm. En la Tabla 2.11 se describen las funciones de ajuste para cada ganancia, especificando los valores de sus parámetros y su cuantía para un 95% del límite de confianza.

**Tabla 2.11. Funciones de ajuste de las ganancias de los sub-sistemas para la herramienta de 10 mm.**

$K_f$ vs $ap_{nom}/n_{nom}$		$K_{ap}$ vs $f_{nom}/n_{nom}$	
$f(x) = p_1x + p_2$		$f(x) = p_1x + p_2$	
$p_1 = 1594$	[624.7, 2563]	$p_1 = 1014$	[666.4, 1363]
$p_2 = -0.167$	[-0.7926, 0.4586]	$p_2 = 80.65$	[23.43, 137.9]
$K_n$ vs $f_{nom}$			
$f(x) = p_1x + p_2$ ;		$p_1 = -0.0005982$	[-0.0009372, -0.0002593]
		$p_2 = 0.02653$	[-0.0478, 0.1009]

**Tabla 2.12. Análisis estadísticos de las constantes de tiempo de los modelos de los sub-sistemas.**

Sub-sistema	Parámetro	Mín.		Máx.		Media		Desviación Standard	
		10 mm	12 mm	10 mm	12 mm	10 mm	12 mm	10 mm	12 mm
$G_f$	$Tp1$	0.111	0.043	0.147	0.181	0.138	0.095	0.0156	0.043
	$Tp2$	0.060	0.014	0.084	0.148	0.069	0.066	0.0101	0.043
	$Td$		0.0034		0.14	0.03	0.03		0.049
$G_n$	$Tp1$	0.0199	0.0039	0.055	0.107	0.035	0.035	0.0123	0.031
	$Tp2$	0.0197	0.0124	0.028	0.052	0.027	0.0312	0.0045	0.016
	$Td$		0.0005		0.03	0.03	0.026		0.013
$G_{ap}$	$Tp1$	0.149	0.115	0.615	0.447	0.343	0.298	0.1164	0.102
	$Tp2$	0.014	0.016	0.148	0.095	0.074	0.040	0.0438	0.025
	$Td$		0.023		0.03	0.03	0.03		0.0020
	$T_{rate}$	2.83	3.37	8.89	8.73	4.01	5.04	1.87	1.35



Por otra parte, en el caso de las constantes de tiempo de los diferentes tipos de modelos, se hizo un análisis estadístico, determinando para cada uno de ellas su valor medio, desviación estándar, el valor mínimo y el máximo. Este análisis se muestra en la Tabla 2.12.

Para el caso del comportamiento temporal del sub-sistema que relaciona la fuerza media de corte con el cambio en la profundidad axial de corte, se hizo un análisis más exhaustivo, planteándose la relación existente entre la mayor constante de tiempo del modelo y el tiempo teórico  $T_{set}$  al cuál debe alcanzarse el estado estacionario en la fuerza ( $T_{rate} = T_{set} / \text{MAX}(Tp1, Tp2)$ ). Los resultados de este análisis se muestran también en la Tabla 2.12. Según trabajos previos [168], se estima que, dado el diámetro de la herramienta  $D$  expresado en milímetros y su velocidad de avance  $f$  dada en mm/min, se alcance el estado estacionario, para el caso de una operación de rasurado, en un tiempo  $T_{set}$ , expresado en segundos, según la ecuación siguiente.

$$T_{set} = \frac{60}{f} \frac{D}{2} \quad (2.33)$$

#### ***Planteamiento generalizado del modelo para ambas herramientas***

A partir del análisis de los parámetros de los modelos obtenidos y previo conocimiento del proceso en estudio se decidió que el modelo dinámico del proceso de FAV, el cual representa los cambios en la fuerza de corte resultante media ante cambios en la velocidad de avance, de giro y la profundidad axial de corte, tuviera la siguiente estructura expresado como función de transferencia basada en el operador de Laplace:

$$F_m(s) = G_f(s)f(s) + G_n(s)n(s) + G_{ap}(s)ap(s) = \frac{K_f}{(T_{f1}s+1)(T_{f2}s+1)} e^{-T_{df}s} f(s) + \quad (2.34)$$

$$+ \frac{K_n}{(T_{n1}s+1)(T_{n2}s+1)} e^{-T_{dn}s} n(s) + \frac{K_{ap}}{T_{ap}s+1} e^{-T_{dap}s} ap(s)$$

Las ganancias del modelo estarán determinadas por las siguientes ecuaciones:

$$K_f = K_{\alpha_{-f-1}} \cdot \frac{ap_{nom}}{n_{nom}} + K_{\alpha_{-f-2}} \quad (2.35)$$

$$K_n = -\frac{f_{nom}}{K_{\alpha_{-n-1}}} + K_{\alpha_{-n-2}} \quad (2.36)$$

$$K_{ap} = K_{\alpha_{-ap-1}} \cdot \frac{f_{nom}}{n_{nom}} + K_{\alpha_{-ap-2}} \quad (2.37)$$

Donde  $f_{nom}$ ,  $n_{nom}$  y  $ap_{nom}$  representan las condiciones nominales de la velocidad de avance, de giro y la profundidad de corte en el momento que se produce el cambio de la variable correspondiente.

De la misma forma, la ecuación (2.38) representa la relación entre la constante de tiempo ante el cambio en la profundidad de corte, el diámetro de la herramienta y la velocidad de avance, para la operación de ranurado:

$$T_{ap} = T_{\alpha_{ap}} \frac{60}{f} \frac{D}{2} \quad (2.38)$$

**Tabla 2.13. Valores de ajuste de los coeficientes para el cálculo de las ganancias del modelo.**

Coeficiente	Unidades	10 mm	12 mm
$K_{\alpha_f_1}$	[N/mm <sup>2</sup> ]	1594	1463
$K_{\alpha_f_2}$	[N/(mm/min)]	-0.167	0
$K_{\alpha_n_1}$	[mm/(N min <sup>2</sup> )]	1671.68	1453.7
$K_{\alpha_n_2}$	[N min]	0.02653	-0.05945
$K_{\alpha_{ap}_1}$	[N/mm <sup>2</sup> ]	1014	1640
$K_{\alpha_{ap}_2}$	[N/mm]	80.65	-36.19
$T_{\alpha_{ap}}$		1/5	

**Tabla 2.14. Valores de ajuste de las constantes de tiempo y retardos de tiempo de los sub-sistemas que conforman el modelo del proceso de FAV.**

Parámetro	10 mm	12 mm
$T_{f1}$	0.1377	0.07
$T_{f2}$	0.0692	0.0439
$T_{df}$	0.03	0.03
$T_{n1}$	0.046	0.0803
$T_{n2}$	0.027	0.0126
$T_{dn}$	0.03	0.03
$T_{dap}$	0.06	0.09

La Tabla 2.13 describe los valores ajustados a los coeficientes para el cálculo de las ganancias del modelo del proceso de FAV, para cada tipo de herramienta, empleando el

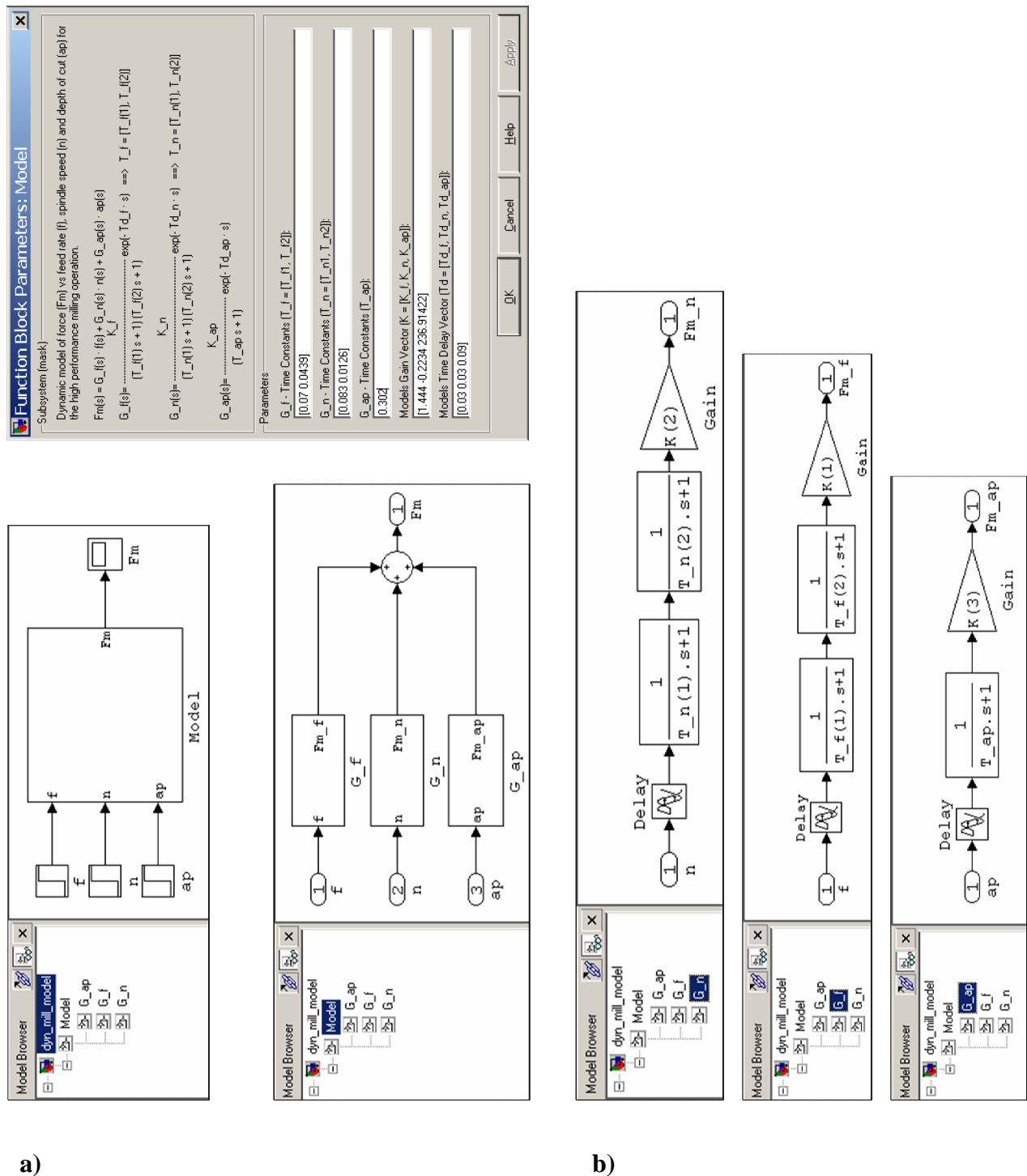
modelo de máquina herramienta de la plataforma experimental y mecanizando el acero de alta dureza 17-4PH. De la misma, bajo esas mismas premisas y a partir del análisis estadístico de la Tabla 2.12, se han ajustado el resto de las constantes de tiempos y retardos de los sub-sistemas que conforman el modelo del proceso de FAV, las cuales son mostradas en la Tabla 2.14.

### 2.4.3 Implementación del modelo en Matlab/Simulink

Con el objetivo de simular el modelo ajustado para su validación y su empleo futuro en el diseño y ajuste de algoritmos de control del proceso de FAV, se ha desarrollado una aplicación en Matlab/Simulink.

La aplicación está compuesta por los sub-sistemas simple entrada-simple salida, correspondientes a cada una de las variables de entrada, y en cada uno de ellos se ha implementado su función de transferencia relacionada. Tanto los polos de cada sub-sistema, como su retardo de tiempo, han sido desarrollados mediante los bloques disponibles en la biblioteca de sistemas continuos de Simulink. Por otra parte, se ha utilizado el algoritmo de Dorman-Price como método numérico para la resolución de ecuaciones diferenciales ordinarias. La aplicación utiliza una sencilla interfaz de usuario para introducir sus parámetros. En la Figura 2.34 se muestran todos los diagramas de bloques que componen el modelo. Las entradas del modelo constituyen cambios en las variables velocidad de avance, velocidad de giro y profundidad axial de corte y su salida los cambios en la fuerza resultante media de corte.

El modelo ha sido simulado con los parámetros ajustados para los dos diámetros de herramienta (10 y 12 mm), empleando las mismas señales registradas de cambios en las variables de entrada pero solamente para sus escalones de bajada, es decir, el intervalo de tiempo  $t_5 \leq t < t_8$  (ver diseño experimental, Figura 2.30 y Tabla 2.8). Debido a que la salida del modelo representa el cambio en la fuerza resultante media ante cambios en sus variables de entrada, para la simulación se han utilizado condiciones iniciales iguales a 0, restando los valores iniciales antes del cambio para cada variable de entrada y el valor inicial de la fuerza medida experimentalmente, para su correcta comparación. En la Figura 2.35 se ilustra el comportamiento del cambio en la fuerza resultante media del modelo y de todas las réplicas de los ensayos realizadas para cada tipo de herramienta, tanto las empleadas para el ajuste del modelo (réplicas 2 y 3 para la herramienta de 10mm y réplicas 1 y 2 para la herramienta de 12mm) como las réplicas para su validación (resto de ensayos).



**Figura 2.34. Desarrollo del modelo dinámico del proceso de FAV en Matlab/Simulink.**  
a) Entrada y salida de datos del modelo e interfaz para introducir sus parámetros y b) diagrama de bloques de los sub-sistemas del modelo.

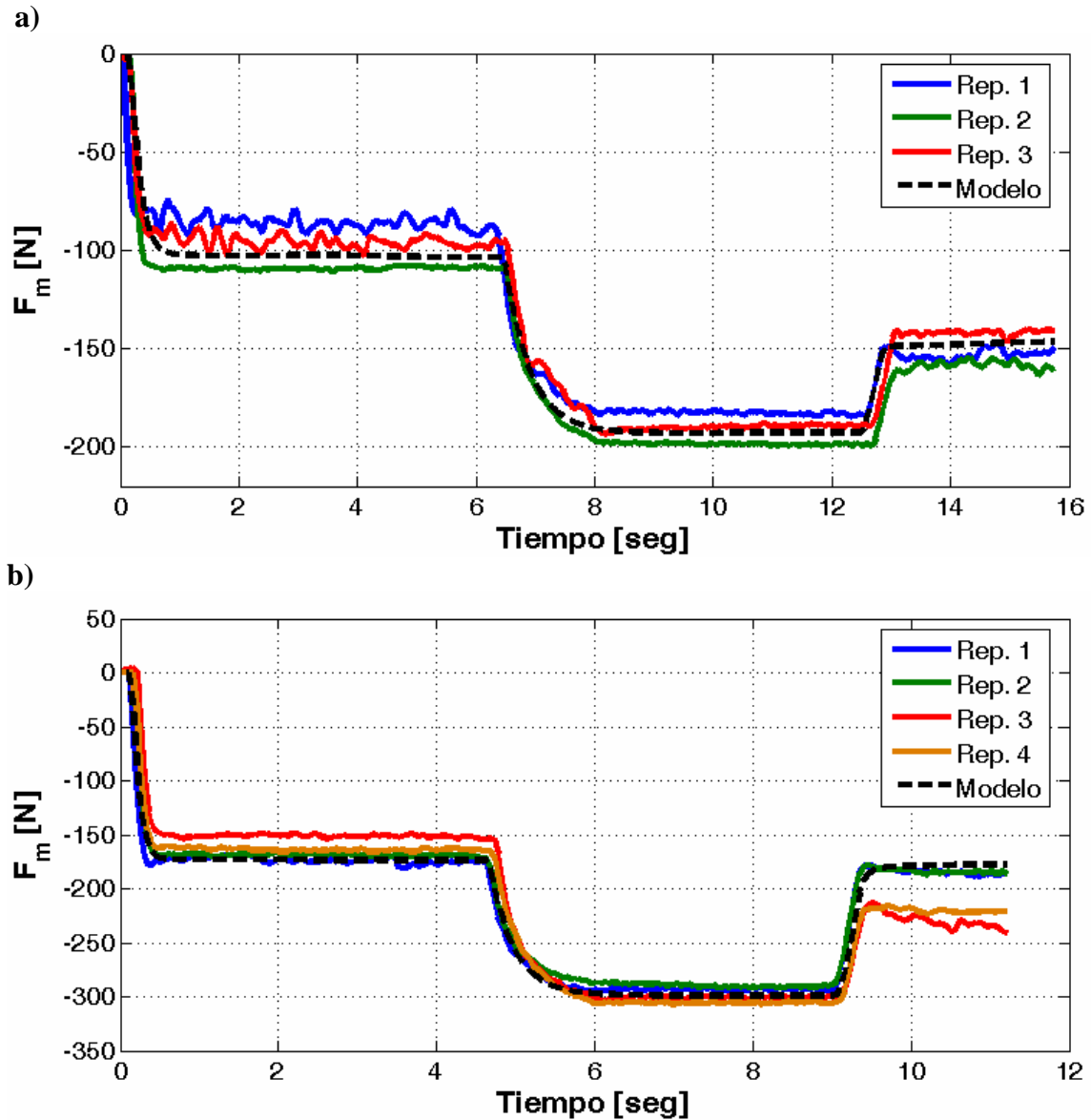


Figura 2.35. Simulación del modelo y comparación con los datos experimentales.  
a) Herramienta de 10 mm y b) herramienta de 12 mm.

### Validación del modelo

El desempeño del modelo implementado en Simulink se ha validado empleando el mismo conjunto de datos de las variables de entrada utilizado para la simulación. Para corroborar la precisión se han calculado dos índices de comportamientos, la integral del error absoluto (IAE), ecuación (2.39), y la integral del error cuadrático (ISE), ecuación (2.40). Para cada muestra  $i$  de los vectores de fuerza, el error se determina como la diferencia entre el valor experimental de la fuerza resultante media  $y(i)$  y la salida del modelo ( $\hat{y}(i) = F_m(i)$ ). En la Tabla 2.15 se relacionan los índices de error calculados para todas las réplicas, tanto las empleadas para ajustar el modelo como las réplicas de validación. Las filas sombreadas se corresponden con las réplicas utilizadas para la validación del modelo. Los índices de error se han calculado en

unidades de fuerza (Newton) y a través de los cuales se puede comprobar el buen desempeño del modelo ajustado e implementado en Simulink.

$$IAE = \frac{1}{N} \sum_{i=1}^N |y(i) - \hat{y}(i)| \quad (2.39)$$

$$ISE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y(i) - \hat{y}(i))^2} \quad (2.40)$$

**Tabla 2.15. Errores del modelo, réplicas de ajuste y de validación (filas sombreadas).**

<b>Diámetro [mm]</b>	<b>Fmax [N]</b>	<b>Réplica No</b>	<b>IAE [N]</b>	<b>(IAE / Fmax) x 100%</b>	<b>ISE [N]</b>	<b>(ISE / Fmax) x 100%</b>
10	200	1	11.59	5.80	13.43	6.72
		2	7.57	3.79	8.82	4.41
		3	5.87	2.94	6.96	3.48
12	300	1	5.09	1.70	6.86	2.29
		2	7.24	2.41	8.87	2.96
		3	20.47	6.82	27.31	9.10
		4	14.58	4.86	19.43	6.48

Por otra parte, en la Tabla 2.15 también se muestra el valor máximo que alcanza el cambio de la fuerza (**Fmax**) durante todo el rango de datos de validación y se ha calculado el valor porcentual de los índices con respecto al valor máximo (quinta y séptima columnas), obteniéndose valores por debajo del 10 % y corroborándose de esta manera la precisión del modelo ajustado.

## 2.5 Conclusiones

En este capítulo se ha presentado un enfoque computacional en el modelado de procesos de fabricación de alta complejidad. Han sido diseñado modelos a partir de ecuaciones físico-matemáticas de los procesos de fabricación y también, basados en la aplicación de técnicas de identificación experimental. Se han desarrollado modelos simple entrada-simple salida (SISO), múltiple entrada-simple salida (MISO) y múltiple entrada-múltiple salida (MIMO). Los modelos han sido implementados mediante aplicaciones desarrolladas en un lenguaje de alto nivel como Matlab/Simulink.

Por otra parte, se ha mostrado la efectividad de los modelos, para la predicción de las fuerzas que se producen en los procesos de fabricación, a través de un estudio comparativo con datos obtenidos experimentalmente. Los modelos computacionalmente eficientes pueden ser utilizados en el diseño, ajuste y optimización de estrategias de control, que a su vez

puedan ser empleadas en sistemas de supervisión y control de procesos de fabricación, con vistas a mejorar la eficiencia de estos procesos.

A pesar de la literatura a la que se ha tenido acceso, en la cual estos procesos han sido objeto de estudio en profundidad y encontrándose disímiles propuestas sobre su modelado, con este capítulo se ha logrado una representación sencilla y útil en un lenguaje de alto nivel, lo que constituye una aportación de la Tesis Doctoral. Por una parte el propio enfoque computacional y los algoritmos desarrollados, constituyen un aporte en sí mismos. Por la otra, las estrategias de diseño para la identificación experimental de ambos modelos dinámicos, el modelo SISO del proceso de taladrado de alto rendimiento y el modelo múltiple entrada-simple salida del proceso de fresado de alta velocidad, son también desde nuestro punto de vista, aportes científicos y técnicos.

En específico, con respecto al modelado del proceso de fresado de alta velocidad, las ecuaciones (2.14) y (2.15) del modelo físico-matemático de las fuerzas de corte, son una contribución. Además, el disponer de un algoritmo computacional que materialice un modelo generalizado del proceso de corte, tiene un valor inestimable para incorporar nuevas capacidades a estos procesos. La implementación de este algoritmo en un sistema jerárquico de monitorización y control en tiempo real, permitiría tanto la posibilidad de detectar anomalías en el proceso, como definir los valores de referencia de las fuerzas de cortes en un sistema de control.

Algunos de los resultados parciales mostrados en este capítulo, cimientos sobre los que se alza la Tesis Doctoral, han sido sometidos a consideración de la comunidad científica, y recibidos positivamente [168, 169]. Por otra parte, y como se verá más adelante, el modelo computacional del proceso de taladrado de alto rendimiento, constituye un componente imprescindible para la metodología propuesta de diseño de estrategias de control óptimas, que serán tratadas en el Capítulo 5. Con vistas a lograr un desarrollo adecuado de esta metodología, se hace necesario conocer el basamento teórico de las estrategias a diseñar, tema que será tratado en el capítulo siguiente.





---

## **CAPÍTULO 3**

# **ESTUDIO DE MÉTODOS EFICIENTES LINEALES, BORROSOS Y NEURO-BORROSOS PARA EL CONTROL DE PROCESOS COMPLEJOS**

---

En este capítulo se realiza un estudio de técnicas clásicas y de *soft-computing* para el control de procesos complejos. Se abordan las ideas esenciales y conceptuales de los sistemas borrosos, los sistemas híbridos neuro-borrosos y las estrategias lineales conocidas, como el algoritmo de acción proporcional, integral y derivativa (PID) para el control de sistemas complejos.

Debido a que estas estrategias han sido tratadas con anterioridad y en profundidad en la literatura científico-técnica, las bases teóricas de las estrategias, así como sus formulaciones matemáticas, son planteadas de forma concisa, haciéndose referencia a la bibliografía más relevante donde se puede hacer una revisión más exhaustiva.

En la primera de las tres secciones en que se divide el capítulo (sección 3.1) se introduce la estrategia lineal de control conocida por su acrónimo como PID, así como algunos de sus métodos de ajuste de mayor uso y se expone de forma muy breve algunas de sus limitaciones en el control de procesos complejos. Seguidamente, en la sección 3.2 se presentan nociones básicas de los sistemas borrosos, así como la estructura y premisas de diseño de los dispositivos lógicos borrosos, ya sea para modelar y/o controlar procesos físicos de una complejidad relativamente alta. Por último, en la sección 3.3, se aborda el estudio de los sistemas híbridos neuro-borrosos como estrategia para la transformación de forma eficiente del conocimiento y la experiencia humana, además del ajuste óptimo de los sistemas de

inferencia borrosa y de esta forma incrementar su desempeño en la monitorización y control de sistemas complejos.

### 3.1 Estudio de estrategias lineales de control (algoritmo PID) y su aplicabilidad al control de procesos complejos

La estrategia lineal de control proporcional, integral y derivativo (PID) y sus versiones, tienen una utilización muy amplia en los sistemas físicos en general. La inmensa mayoría de los procesos continuos, contienen sistemas de control realimentados gobernados por un controlador PID, ya sea implementado mediante un dispositivo mecánico, mediante un circuito electrónico analógico o mediante un dispositivo de control digital con una versión discreta del algoritmo.

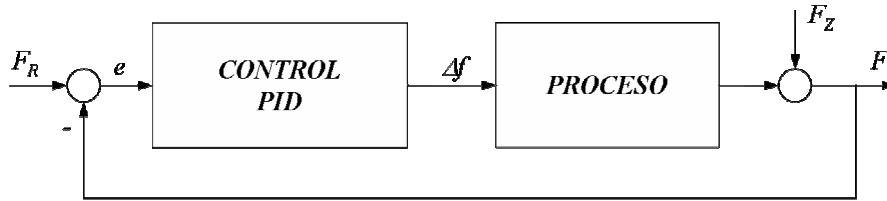


Figura 3.1. Diagrama general de un controlador PID en un sistema de control realimentado.

La Figura 3.1 muestra un diagrama de bloques de un sistema de control a lazo cerrado, simple entrada-simple salida, con una estrategia PID como controlador. La función teórica en el dominio del tiempo, de la versión clásica del controlador, la cual define su relación entrada-salida, se expresa como:

$$\Delta f(t) = K_p \cdot \left( e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (3.1)$$

siendo,  $t$  el tiempo,  $e(t)$  la señal de error (entrada) entre el valor de referencia del sistema de control y la salida del proceso ( $e(t) = F_R(t) - F(t)$ ),  $\Delta f(t)$  la señal de control (salida del controlador) y los tres parámetros básicos del controlador dados por su ganancia proporcional  $K_p$ , la constante de tiempo integral  $T_i$  y la constante de tiempo derivativa  $T_d$ .

La ecuación del controlador puede expresarse también como la suma de tres términos: la acción proporcional, la acción integral y la derivativa

$$\begin{aligned}
\Delta f(t) &= \Delta f_P(t) + \Delta f_I(t) + \Delta f_D(t) \\
\Delta f_P(t) &= K_P e(t) \\
\Delta f_I(t) &= K_I \int_0^t e(\tau) d\tau \\
\Delta f_D(t) &= K_D \frac{de(t)}{dt}
\end{aligned} \tag{3.2}$$

donde  $K_I = K_P / T_I$  y  $K_D = K_P T_D$ .

Se puede hacer una representación por medio de la Transformada de Laplace, y expresar mediante una función de transferencia con el operador de Laplace ( $s$ ):

$$G_{PID}(s) = \frac{\Delta f(s)}{e(s)} = K_P + \frac{K_I}{s} + K_D \cdot s \tag{3.3}$$

Igualmente, mediante ecuaciones en diferencias y de forma discreta, considerando un tiempo de muestreo  $T_s$ , se puede representar como:

$$\begin{aligned}
\Delta f[k] &= \Delta f_P[k] + \Delta f_I[k] + \Delta f_D[k] \\
\Delta f_P[k] &= K_P e[k] \\
\Delta f_I[k] &= \Delta f_I[k-1] + (K_I T_s) e[k] \\
\Delta f_D[k] &= (K_D / T_s) (e[k] - e[k-1])
\end{aligned} \tag{3.4}$$

Debido a que la acción derivativa del controlador es sensible a los cambios de la señal de error, el ruido que puede existir en esta señal, algo muy común en los procesos físicos reales, puede afectar el comportamiento del sistema de control. Por tanto una versión del controlador, ampliamente utilizada, añade un filtro pasa-bajo en la componente derivativa:

$$\Delta f_D[k] = \frac{T_d}{T_d + N \cdot T_s} \Delta f_D[k-1] + \frac{K_P \cdot T_d \cdot N}{T_d + N \cdot T_s} \cdot (e[k] - e[k-1]) \tag{3.5}$$

siendo  $N$  el coeficiente del filtro.

La Figura 3.2 muestra, a modo de ejemplo, una implementación del algoritmo de control en Matlab/Simulink, el cual puede ser utilizado para tareas de simulación o controlar directamente un proceso a través de la herramienta Real-Time Workshop (RTW), también disponible en el entorno de Matlab.

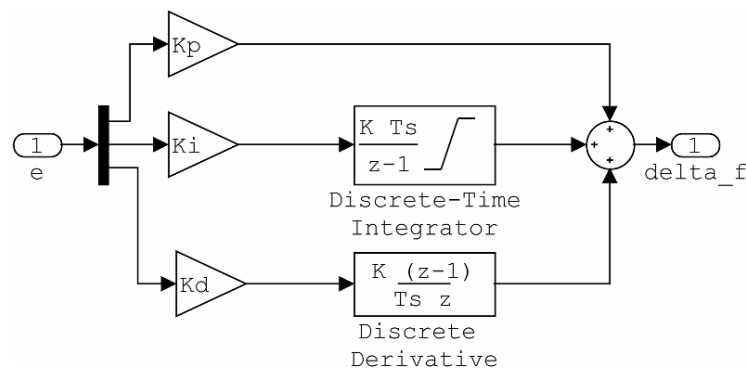


Figura 3.2. Diagrama en Matlab/Simulink del controlador PID discreto.

### 3.1.1 Métodos de ajuste

El ajuste de los parámetros del controlador es relativamente fácil e intuitivo [170]. Existen varios métodos para su ajuste que, en general, parten de un modelo lineal del proceso conocido. Algunos de los métodos de ajuste más conocidos, aparte del método manual por prueba y error, son el de Ziegler-Nichols y Cohen-Coon. Una revisión más profunda de estos temas está fuera del ámbito de esta Tesis Doctoral.

Sobre el método de Ziegler-Nichols existen dos versiones, la más conocida de ellas basada en el análisis en el dominio frecuencial. Esta variante consiste en obtener la ganancia crítica  $K_u$  (ganancia límite del controlador sólo con acción proporcional) con la cual se producen oscilaciones sostenidas en el sistema de control, siendo  $T_u$  el periodo de dichas oscilaciones. A partir de estos valores se ajustan los parámetros del controlador.

Tabla 3.1. Métodos de ajuste de algoritmos de control PID.

Parámetros Métodos	$K_p$	$T_i$	$T_d$
Ziegler-Nichols – dominio frecuencial	$0.6 \cdot K_u$	$0.5 \cdot T_u$	$0.125 \cdot T_u$
Cohen-Coon $a = K_1 \frac{L}{T_c}, \tau = \frac{L}{L+T_c}$	$(1.35/a) \cdot \left(1 + 0.18 \cdot \frac{\tau}{1-\tau}\right)$	$L \cdot \frac{2.5 - 2 \cdot \tau}{1 - 0.39 \cdot \tau}$	$L \cdot \frac{0.37 - 0.37 \cdot \tau}{1 - 0.81 \cdot \tau}$
Visioli (procesos con retardo de transporte $L$ y ganancia a estado estacionario $K_1$ )	$1.37 / (K_1 \cdot L)$	$1.49 \cdot L$	$0.59 \cdot L$

Por otra parte, el método de Cohen-Coon está basado en un modelo del proceso de primer orden y con retraso de transporte (3.6), y tiene como objetivo principal el rechazo a perturbaciones de un sistema de control. Otro de los métodos a resaltar de uso más reciente es el de Visioli [171], basado también en un modelo lineal del proceso con retraso de transporte. Su objetivo primario es la minimización de la integral del error ya sea ante presencia de perturbaciones o ante cambios en la variable de referencia del sistema de control. En la Tabla 3.1 se resumen los tres métodos de ajustes, así como las ecuaciones para el ajuste de los parámetros del controlador PID.

$$G_P = \frac{K_1}{1 + sT_c} e^{-sL} \quad (3.6)$$

### 3.1.2 Algunas limitaciones

A pesar de que la estrategia de control PID es aplicable a muchos problemas de control en la industria, teniendo un desempeño muy satisfactorio, aún sin un ajuste preciso de sus parámetros, en algunas aplicaciones no es capaz de proveer un control óptimo. Su principal problema reside en que sus parámetros son constantes y por tanto su curva o superficie de respuesta estática es lineal. Esto hace que ante no linealidades e incremento de la complejidad en procesos físicos, su desempeño se empobrezca. Su rendimiento puede mejorarse aplicando algunas estrategias, como la introducción en el sistema de control de acciones anticipativas, algoritmos de planificación de ganancias que cambian los parámetros del controlador bajo diferentes condiciones de operación, entre muchas otras.

## 3.2 Estudio de los sistemas borrosos y su aplicabilidad al control de procesos complejos

Los diseñadores de los sistemas de control han buscado incesantemente nuevos caminos para la solución al problema de la regulación de procesos complejos. Esto es especialmente así en los casos en que *no existe un modelo preciso del proceso y la información previa acerca de éste es de carácter esencialmente cualitativo*. El problema del control constituye un caso típico de toma de decisiones. Es por ello que la creación de la lógica borrosa (*fuzzy*) por Lofti Zadeh a inicios de los 60, significó la aparición de una nueva y poderosa herramienta para hacer frente a situaciones de *imprecisión esencial* en modelos, información, objetivos, restricciones y acciones de control. Ya en 1923 Bertrand Russell, en su artículo *Vagueness* expresaba que “[...] la lógica tradicional supone el uso de conceptos precisos, que, sin embargo, son sólo aplicables en un escenario ideal...” [172]. El propio Zadeh estableció el llamado *principio de incompatibilidad*: “Si crece la complejidad del sistema, decrece nuestra aptitud para formular juicios precisos y significativos sobre su comportamiento, hasta

alcanzar una frontera, tras la cual la precisión y la pertinencia son prácticamente características mutuamente incompatibles” [173].

En el campo específico de los sistemas de control, es frecuente el caso de procesos en los que, las estrategias de control diseñadas según otros patrones (clásicos, adaptativos, etc.), no ofrecen los resultados deseados o simplemente fallan del todo. Se trata de procesos de elevada complejidad y a pesar de ello, en la inmensa mayoría de los casos, los operadores humanos diestros y experimentados alcanzan resultados satisfactorios. En 1965, Zadeh da a la luz la llamada teoría de conjuntos borrosos, que echa los cimientos de la llamada síntesis lingüística, mostrando cómo pueden usarse planteos lógicos vagos para derivar inferencias (también vagas) a partir de datos imprecisos [174]. Aunque este último trabajo sugiere aplicaciones a sistemas humanísticos, en realidad la extensión a la regulación automática de procesos se hizo evidente. Al menos así lo fue para Ebrahim Mamdani [23] y su grupo del Queen Mary College de Londres y para su contemporáneo Richard M. Tong [175], de la cercana Universidad de Cambridge. Fue en efecto un numeroso colectivo de científicos de varias nacionalidades europeas, que incluyó a Assilian, Ostergaard, Kickert, King, Van Nauta Lemke, Baaklini, Procyk, Rutherford y Pappis, entre otros, el de los pioneros en la aplicación de la lógica borrosa al control, a partir de la segunda mitad de los 70.

Junto con la aparición de la lógica borrosa, surgieron los primeros trabajos que intentaban caracterizar el funcionamiento del regulador borroso [176]. Sin embargo, uno de los trabajos que marca un hito en esta temática fue el desarrollado por Chuen Chien Lee, en una época donde ya se habían consolidado las aportaciones teóricas y prácticas en este campo [177]. En esta publicación aparece una interesante lista de los trabajos que, desde la óptica del autor, representan los 17 hitos más importantes en el desarrollo del control borroso, dados a la luz entre 1972 y 1988. De entre éstos, se resaltan algunos de los más importantes:

- Los trabajos pioneros (Zadeh, Mamdani)
- La primera aplicación industrial a gran escala y de elevadas prestaciones (Control del horno de cemento en Oregón desarrollado por Holmblad y Ostergaard)
- La introducción de reglas con consecuente determinista (reglas “tipo Sugeno”)
- El desarrollo de “hardware” a partir de circuitos especializados.
- Las masivas aplicaciones, con la industria japonesa a la cabeza.

Vale la pena comentar que en la actualidad, a pesar de gozar de una aceptación generalizada, el reconocimiento de la validez, rigor y utilidad de la teoría de los conjuntos borrosos no ha estado exento de controversias. Algunos científicos de renombre han llegado a lanzarle fuertes inyectivas, imputándole ingenuidad filosófica, carencia de utilidad e interés, excepto el puramente matemático [178, 179].

Sin embargo, las aplicaciones generalizadas y el desarrollo teórico de los últimos 20 años, se han encargado de darle a esta teoría su justo valor, probablemente menos “milagrosa” de como nos la describiera Zadeh al principio, pero indudablemente muy útil y poderosa.

A día de hoy, es muy difícil de refutar, que la lógica borrosa ofrece una alternativa adicional, una herramienta más, para la solución técnica al problema de toma de decisiones y al control inteligente de procesos, es decir, que ha venido a enriquecer el arsenal a disposición de los diseñadores de sistemas de monitorización y control, con un enfoque muy novedoso para emular la toma de decisiones humana. La idea de incorporar al regulador la experiencia de los operadores expertos y su forma de actuar sobre el proceso, le proporciona el “conocimiento” necesario para hacer frente a la complejidad y severas no linealidades de muchos sistemas.

### 3.2.1 Algunas definiciones básicas

A continuación se resumen algunos de los conceptos básicos de la Teoría de los Conjuntos Borrosos y la lógica borrosa, que contribuirán a una mejor comprensión del trabajo. Más detalles se pueden encontrar en los trabajos de Kaufmann [180], Zimmermann [181], Dubois y Prade [182], entre otros.

Sea pues  $U$  un *conjunto clásico* de objetos, conocido como *universo de discurso* (o conjunto de referencia), cuyos elementos genéricos se representan como  $x$ . La pertenencia a un sub-conjunto  $A$  de  $U$  podría verse como una cierta función característica  $\mu_A$  de  $U \rightarrow \{0,1\}$  (Figura 3.3a) tal que

$$\begin{aligned}\mu_A(x) &= 1 \Leftrightarrow x \in A \\ \mu_A(x) &= 0 \Leftrightarrow x \notin A\end{aligned}\tag{3.7}$$

Lo anterior no es más que un planteamiento formal de la expresión: “El objeto  $x$  pertenece (ó no) al sub-conjunto  $A$ ”. Al par  $\{0,1\}$  se le conoce como conjunto de evaluación.

En la lógica borrosa este conjunto se transforma en el intervalo real  $[0,1]$ ,  $A$  se denomina *sub-conjunto borroso* (Figura 3.3b) y  $\mu_A(x)$  es el grado de pertenencia de  $x$  a  $A$ . Mientras más cercano a  $1$  sea el valor de  $\mu_A(x)$ , mayor será la pertenencia. Así,  $A$  resulta ser un sub-conjunto de  $U$  con fronteras no abruptas, por lo que un sub-conjunto borroso puede verse como un predicado cuyos valores **ciertos** (verdaderos) se adquieren del intervalo  $[0,1]$ .

Entonces,  $A$  está completamente caracterizado por el conjunto de pares ordenados

$$A = \{(x, \mu_A(x)), x \in U\}\tag{3.8}$$

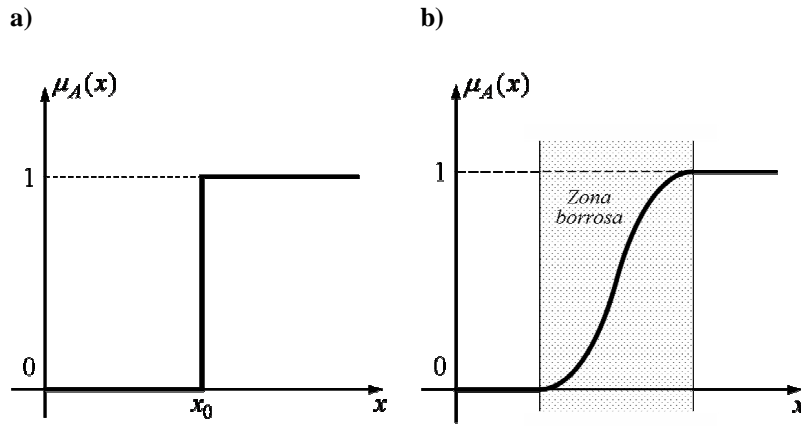


Figura 3.3. Sub-conjunto clásico y borroso.

a) Sub-conjunto clásico con frontera abrupta en  $x_0$ , b) sub-conjunto borroso.

Para el caso de universos discretos  $U = \{x_1, \dots, x_n\}$ , una notación quizá más conveniente sugerida por Zadeh es

$$A = \mu_A(x_1)/x_1 + \dots + \mu_A(x_n)/x_n = \sum_{i=1}^n \mu_A(x_i)/x_i \quad (3.9)$$

Extendiéndola para  $U$  no finito,

$$A = \int_U \mu_A(x)/x \quad (3.10)$$

El *soporte* de un conjunto borroso  $A$  es el sub-conjunto ordinario de  $U$ , tal que

$$\text{supp } A = \{x \in U, \mu_A(x) > 0\} \quad (3.11)$$

Un *corte- $\alpha$*  (o corte-nivel) de un conjunto borroso  $A$ , es el sub-conjunto ordinario de  $U$  tal que

$$A_\alpha = \{x \in U, \mu_A(x) \geq \alpha\} \quad (3.12)$$

Los elementos de  $x \in U$  tales que  $\mu_A(x) = 0.5$  constituyen los puntos de cruce de  $A$  y la altura de  $A$ , por ejemplo, la cota mínima superior de  $\mu_A(x)$ , se define como:

$$\text{hgt}(A) = \sup_{x \in U} \mu_A(x) \quad (3.13)$$

Se dice que  $A$  está normalizado si y solo si existe al menos una  $x \in U$ , tal que  $\mu_A(x) = 1$ . El conjunto vacío  $\emptyset$  se define como  $\mu_\emptyset(x) = 0, \forall x \in U$ , claro que  $\mu_U(x) = 1, \forall x \in U$ . Una



singularidad borrosa (*fuzzy singleton*) es un conjunto borroso puntual, o sea,  $\mu_s(x) \neq 0, \forall x \in U$  y frecuentemente  $\mu_s(x_0) = 1$ , aunque por lo general  $0 < \mu_s(x_0) \leq 1$  (Figura 3.4). Otras definiciones que pudieran resultar interesantes, pueden hallarse en la literatura, como por ejemplo en el trabajo de Dubois y Prade [182].

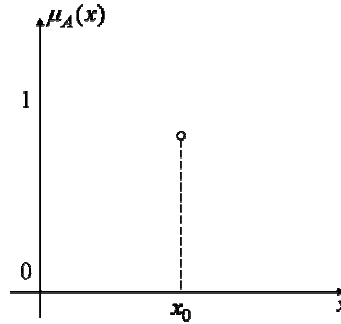


Figura 3.4. Conjunto borroso puntual o singularidad borrosa.

### 3.2.2 Operaciones básicas de la lógica borrosa

De igual modo que para los conjuntos convencionales, para los conjuntos borroso también se definen las operaciones de unión e intersección, que por demás se emplean con mucha frecuencia en control borroso. La primera definición, conocida como min-max, fue formulada por Zadeh. A los respectivos conjuntos borrosos les corresponderían funciones de pertenencia:

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad \forall x \in U \quad (3.14)$$

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad \forall x \in U \quad (3.15)$$

Suele usarse para estas operaciones la siguiente simbología:

- $A \vee B$  ó  $\mu_A \vee \mu_B$  para el máximo,
- $A \wedge B$  ó  $\mu_A \wedge \mu_B$  para el mínimo.

La justificación estrictamente matemática de la elección de las funciones para la unión e intersección, ha sido estudiada por varios autores, considerando en general dos operadores  $f$  y  $g$ . Las definiciones de (3.7) y (3.15), dan lugar a las siguientes propiedades de  $f$  y  $g$ :

1. La pertenencia de  $x$  al conjunto borroso resultante de las operaciones depende exclusivamente de las funciones de pertenencia de los operandos.
2.  $f$  y  $g$  son conmutativos, asociativos e idempotentes, o sea (para  $g$  vale lo mismo que para  $f$ )

3.  $f$  y  $g$  son no decrecientes, o sea que la pertenencia de  $x$  a  $A \cup B$  ó a  $A \cap B$  no puede decrecer cuando la pertenencia de  $x$  a  $A$  ó a  $B$  crezca.
4.  $f(u, u)$  y  $g(u, u)$  son estrictamente crecientes. Si  $\mu_A(x_1) = \mu_B(x_1) > \mu_A(x_2) = \mu_B(x_2)$ , entonces la pertenencia de  $x_1$  a  $A \cup B$  ó a  $A \cap B$  es ciertamente mayor que la de  $x_2$ .
5. La pertenencia a  $A \cap B$  requiere de más y la pertenencia a  $A \cup B$  de menos que la pertenencia a  $A$  ó  $B$  individualmente y por separado.

$$\begin{aligned}\mu_{A \cup B}(x) &\geq \max(\mu_A(x), \mu_B(x)) \\ \mu_{A \cap B}(x) &\leq \min(\mu_A(x), \mu_B(x))\end{aligned} \quad \forall x \in U \quad (3.16)$$

6. La pertenencia total a  $A$  y a  $B$  implica pertenencia completa a  $A \cap B$ . La ausencia total de pertenencia a  $A$  y a  $B$  implica total ausencia de pertenencia a  $A \cup B$ .

$$g(1, 1) = 1; f(0, 0) = 0 \quad (3.17)$$

7.  $f$  y  $g$  pueden extenderse, recursivamente, a  $m \geq 3$  argumentos.
8.  $\forall x \in U, f(1, \mu_A(x)) = 1; g(0, \mu_A(x)) = 0$

Por otra parte, al complemento  $\bar{A}$  de  $A$  le corresponde la función de pertenencia

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad \forall x \in U \quad (3.18)$$

Un rasgo significativo que distingue a los conjuntos borrosos de los clásicos es que, para conjuntos borrosos, en general  $\bar{A} \cap A \neq \emptyset$ ;  $\bar{A} \cup A \neq U \quad \forall A \in U$ .

Aún más, en general, si  $A$  es cierto concepto o idea, se dice que tal concepto es borroso cuando no se distingue claramente entre  $\bar{A}$  y  $A$ .

Aunque los útiles diagramas de Venn no existen para los conjuntos borrosos, en el sentido de cómo son utilizados para sub-conjuntos ordinarios, pueden usarse los gráficos de las funciones de pertenencia para visualizar las operaciones más arriba definidas. Sin embargo, las definiciones para las operaciones de unión e intersección borrosas no son únicas. Otros operadores empleados, similares a sus homólogos probabilísticos, son:

- Para la unión, conocida como suma algebraica ó probabilística:

$$\mu_{A \dot{+} B}(x) = f'(\mu_A(x), \mu_B(x)) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \quad \forall x \in U \quad (3.19)$$

- Para la intersección, denominada producto ó producto algebraico:

$$\mu_{A \cdot B}(x) = g'(\mu_A(x), \mu_B(x)) = \mu_A(x) \cdot \mu_B(x) \quad \forall x \in U \quad (3.20)$$

Las funciones  $f'$  y  $g'$  satisfacen la conmutatividad y la asociatividad, además de la identidad,  $A \cup \emptyset = A$ ;  $A \cap A = A$ , las Leyes de Morgan  $\overline{(A \cap B)} = \bar{A} \cup \bar{B}$ ;  $\overline{(A \cup B)} = \bar{A} \cap \bar{B}$  y  $A \cdot \emptyset = \emptyset$ ;  $A \hat{+} U = U$ . Para estos operadores también pueden construirse diagramas de Venn (ver Figura 3.5).

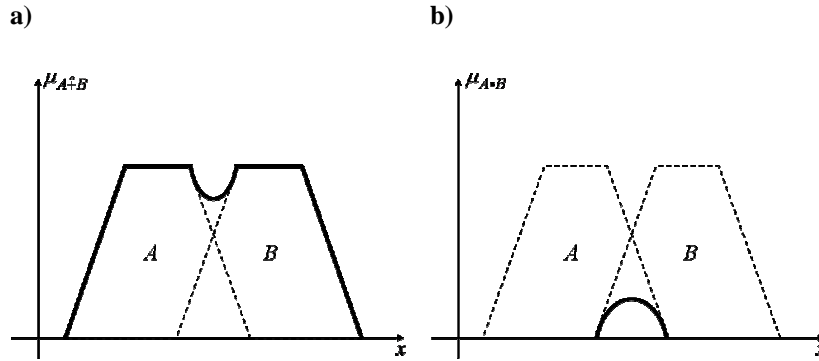


Figura 3.5. Diagramas de Venn.  
a) Operador unión y b) operador intersección.

### 3.2.3 Producto Cartesiano

En la teoría de conjuntos clásica se denomina producto Cartesiano de dos clases  $U$  y  $V$ , a otra clase

$$U \times V = \{ \langle x, y \rangle, x \in U \text{ y } y \in V \} \quad (3.21)$$

La definición de tal operación en la lógica borrosa es como sigue. Si  $A_1, \dots, A_n$  son conjuntos borrosos en los universos  $U_1, \dots, U_n$ , respectivamente, el producto Cartesiano de  $A_1 \times \dots \times A_n$  es un conjunto borroso en el espacio-producto  $U_1 \times \dots \times U_n$ , con función de pertenencia

$$\mu_{A_1 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \min \{ \mu_{A_1}(x_1), \dots, \mu_{A_n}(x_n) \} \quad (3.22)$$

ó, en su lugar

$$\mu_{A_1 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \mu_{A_1}(x_1) \cdot \mu_{A_2}(x_2) \cdot \dots \cdot \mu_{A_n}(x_n) \quad (3.23)$$

Nótese que en el producto Cartesiano se trata de  $n$  conjuntos borrosos y funciones de pertenencia de otros tantos *universos diferentes*. Una representación gráfica simple, no es posible para el producto Cartesiano. Esto se debe a que, más que una curva, el conjunto borroso  $\mu_{A_1 \times \dots \times A_n}(x_1, x_2, \dots, x_n)$  generaría una hipersuperficie, en un hiperespacio de  $(n+1)$

dimensiones, donde, a cada vector  $[\mu_{A_1}(x_1) \cdots \mu_{A_n}(x_n)]^T$  le corresponde un punto, cuya restante coordenada se calcula por medio de (3.22) ó de (3.23). El tema de las operaciones lógicas borrosas se retomará más adelante, al abordar la lógica de toma de decisiones del controlador borroso.

### 3.2.4 Composición “Sup-\*”

Si  $R$  y  $S$  son relaciones borrosas en  $U \times V$  y  $V \times W$  respectivamente, la composición de  $R$  y  $S$  es una relación borrosa denotada por  $R \circ S$  y se define por:

$$R \circ S = \left\{ \left[ (u, w), \sup_v (\mu_R(u, v) * \mu_S(v, w)) \right], u \in U, v \in V, w \in W \right\} \quad (3.24)$$

Donde “\*” puede ser cualquier operador perteneciente a la clase de las llamadas “**normas-T**” (mínimo, producto algébrico, producto acotado, producto drástico) [183].

De forma tal que la regla de inferencia composicional “Sup-\*” se define como:

Si  $R$  es una relación borrosa en  $U \times V$  y  $x$  es un conjunto borroso en  $U$  entonces la regla composicional de inferencia “Sup-\*” afirma que el conjunto borroso  $y$  en  $V$ , inducido a partir de  $x$  viene dado por [184]:

$$y = x \circ R \quad (3.25)$$

donde  $x \circ R$  es la composición “Sup-\*” de  $x$  en  $R$ . Si “\*” representa el operador mínimo entonces esta definición se reduce a la regla de inferencia “Sup-Min” de Zadeh [173].

### 3.2.5 Variable lingüística. Definición

Una definición, equivalente en cierta medida a la de partición borrosa, es la de variable lingüística [185]. Una variable lingüística está caracterizada por una quintupla  $(x, T(x), U, G, M)$ , en la que,

- $x$ : nombre de variable,
- $T(x)$ : conjunto de valores lingüísticos (atributos, adjetivos) de  $x$ ,
- $U$ : universo de discurso,
- $G$ : regla sintáctica para generar los nombres de  $x$ ,
- $M$ : regla semántica para asociar cada valor a su significado.

Por ejemplo, si la *fuerza* es interpretada como una variable lingüística  $x$ , entonces su conjunto de valores lingüísticos, es decir la *partición* de su universo, se elegirá de entre: baja, moderada, media, alta, muy alta, etc. Un caso sería tal y como se muestra en la Figura 3.6.

$$T(fuerza) = T(x) = \{baja, media, alta\}$$

Según la partición elegida, el uso de una regla sintáctica no sería tan necesario. Otro fuera el caso si el conjunto de atributos no tuviera un significado tan explícito, por ejemplo en

$$T(x) = \{\dots, \text{bastante alta}, \text{excesivamente alta}, \text{muy alta}, \dots\}$$

Entonces se hace imprescindible el disponer de una regla para ordenar la ubicación de estos atributos en el universo de discurso. Por ejemplo

G:..., “muy alta” en el extremo superior de  $U$ , a continuación “excesivamente alta” y le sigue “bastante alta”.

$$M_{ALTA} = \begin{cases} \mu_{ALTA} = 1, & F \geq 2000N \\ \mu_{ALTA} = \frac{F}{500} - 3, & 1500N \leq F \leq 2000N \\ \mu_{ALTA} = 0, & F \leq 1500N \end{cases}$$

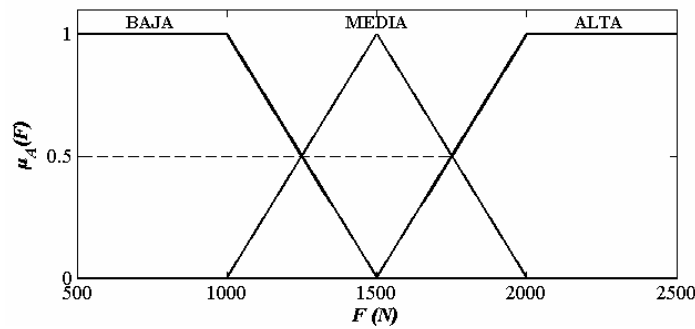


Figura 3.6. Variable lingüística fuerza.

La regla semántica  $M$  permitiría entonces precisar, en nuestro ejemplo, que “las fuerzas se califican plenamente (100%) como “altas” por encima de los 2000  $N$ , valor de fuerza en la que la pertenencia a este conjunto decae linealmente hasta ser nula en 1500  $N$ ”. O sea

### 3.2.6 El dispositivo lógico borroso: premisas de diseño

Según los enfoques tradicionales, el diseño de un controlador se basa en un análisis profundo del proceso y una vez que se posee un modelo cuantitativo de éste, todas las decisiones se calculan usando algoritmos estrictamente numéricos. Es obvio que sólo a procesos “dóciles” al análisis cuantitativo les son aplicables técnicas de control de entidad numérica. Por otra parte, como se apuntara con anterioridad, existe un número de procesos con cierta complejidad y por tanto difíciles de controlar automáticamente, debido a su modelado preciso, pero que sin embargo, los operadores humanos exhiben magníficos resultados en su regulación. Esto, junto a la aparición de la teoría de los conjuntos borrosos y la disponibilidad de medios de cómputo poderosos, estimuló la investigación acerca de las

estrategias de toma de decisiones de esos operadores, expresadas por medio de reglas heurísticas, para convertirlas en estrategias de control automático.

Según lo explicado, el dispositivo lógico borroso (en lo adelante DLB) resultante se basaría en la estrategia del operador humano, o sea, en un modelo de toma de decisiones del experto. La esencia de tal modelo es un programa basado en reglas, por lo que clasifica entre los llamados sistemas expertos y tiene casi siempre un carácter de modelo lingüístico (al menos es así en el llamado control borroso clásico). Las reglas a que nos referimos ahora toman la forma

$$SI [precedencia] ENTONCES [consecuencia]$$

La precedencia refleja el estado del proceso, en principio, mediante variables lingüísticas como las antes definidas y la consecuencia representa las acciones de control o variables manipuladas (entradas del proceso), también lingüísticas, que se les hacen corresponder por el experto. De modo que la precedencia sería de la forma

$$\left[ \begin{array}{c} Variable\ de \\ estado\ 1 \end{array} \right] ES \left[ \begin{array}{c} Conjunto \\ borroso \end{array} \right] Y \left[ \begin{array}{c} Variable\ de \\ estado\ 2 \end{array} \right] ES \left[ \begin{array}{c} Conjunto \\ borroso \end{array} \right] Y \dots$$

y la consecuencia

$$\left[ \begin{array}{c} Variable\ de \\ acción\ 1 \end{array} \right] ES \left[ \begin{array}{c} Conjunto \\ borroso \end{array} \right] ; \left[ \begin{array}{c} Variable\ de \\ acción\ 2 \end{array} \right] ES \left[ \begin{array}{c} Conjunto \\ borroso \end{array} \right] ; \dots$$

A modo de ejemplo, una de las también llamadas reglas de producción podría ser:

$$SI [error] ES [alto] Y [delta\ error] ES [muy\ alto] ENTONCES [velocidad\ de\ avance] ES [disminuir]; [velocidad\ de\ giro] ES [reducir\ mucho]$$

Se observa que las premisas (precedencia) están relacionadas entre sí mediante el conectivo “Y”, mientras que las consecuencias están separadas con “;”. Se llama la atención en este sentido, pues algunos autores usan para la consecuencia “Y” en lugar de “;”. Esto puede acarrear errores de interpretación, porque su significado es bien diferente a cuando se usa en la precedencia y así ha de reflejarse a la hora de darles expresión formal para realizar los cálculos. De la investigación acerca de la estrategia del operador sale un conjunto de reglas como éstas:

$$\vdots \\ R_i : \left[ \begin{array}{c} variable\ de \\ estado\ 1 \end{array} \right] ES \left[ \begin{array}{c} conjunto \\ borroso \end{array} \right] Y \dots ENTONCES \left[ \begin{array}{c} variable\ de \\ acción\ 1 \end{array} \right] ES \left[ \begin{array}{c} conjunto \\ borroso \end{array} \right] ; \dots$$

$TAMBIÉN$   
 $R_{i+1} : \begin{bmatrix} \text{variable de} \\ \text{estado 1} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix} Y \dots ENTONCES \begin{bmatrix} \text{variable de} \\ \text{acción 1} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix}; \dots$   
 $TAMBIÉN$   
 $\vdots$

Todas las reglas del protocolo son, en principio, válidas y aplicables, lo que explica la presencia del conectivo “*TAMBIÉN*”. Claro que, ante cada estado del proceso, la validez y efectividad de cada regla será “modulada” por el grado de cumplimiento de sus respectivas premisas. La teoría de conjuntos borrosos debe proveer entonces las herramientas matemáticas necesarias para crear los correspondientes algoritmos de cálculo.

### 3.2.7 Configuración básica del DLB

La configuración básica más universalmente aceptada para representar, en forma general, el procedimiento antes descrito, es la mostrada en la Figura 3.7 [186], donde el DLB realiza una función de controlador y aparece conectado con el proceso en un lazo cerrado de control.

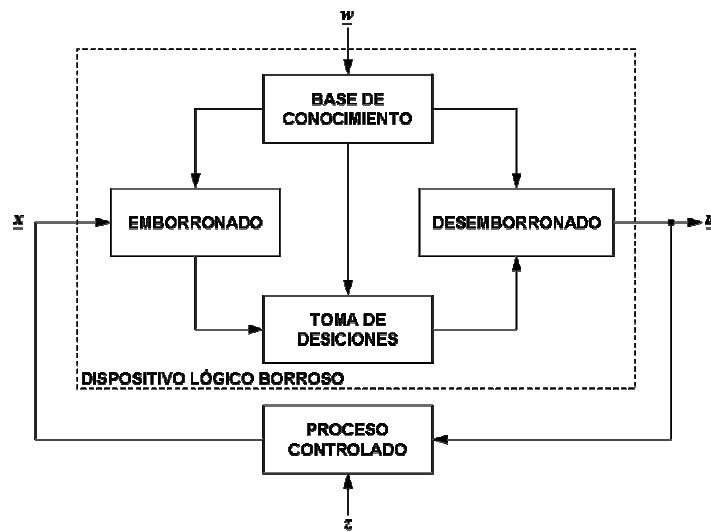


Figura 3.7. Diagrama en bloques del lazo de control con DLB.  
(Haber, 1998 [186])

En ella, la entrada al controlador es el vector  $\underline{x}$  de las variables de estado del proceso y su salida, el conjunto de variables de acción correctiva  $\underline{u}$ , generadas por el DLB. Los vectores  $\underline{w}$  y  $\underline{z}$  representan la información introducida desde el sistema y las perturbaciones que actúan sobre el proceso, respectivamente.

El DLB maneja cada una de los componentes de ambos vectores como variables lingüísticas. De modo que, para una cualquiera de las variables de estado  $x_i$  y acción  $u_h$ , los conjuntos de atributos serían, respectivamente:

$$\begin{aligned} T(x_i) &= \{A_1^i, A_2^i, \dots, A_{m_i}^i\}; \quad i = 1, 2, \dots, n \\ T(u_h) &= \{B_1^h, B_2^h, \dots, B_{q_h}^h\}; \quad h = 1, 2, \dots, \tilde{n} \end{aligned} \quad (3.26)$$

donde  $A_j^i, B_l^h$  son conjuntos borrosos,  $m_i$  es el número de conjuntos borrosos en que se parte  $x_i$  y  $q_h$  el número de conjuntos borrosos en que se parte  $u_h$ .

El bloque de emborronado sirve de interfaz entre la evaluación determinista de las variables de estado y la correspondiente expresión borrosa. Incluye la medición de las variables de entrada, eventual normalización de los valores de dichas variables en una gama preestablecida y su conversión a valores lingüísticos.

La base de conocimiento comprende, en esencia, el conocimiento de la esfera de aplicación y las metas de control del operador, dadas por un conjunto de reglas (implicaciones) como las mostradas. La precedencia y consecuencia suelen simbolizarse con mayúsculas en negrillas, de modo que el conjunto de las reglas podría expresarse globalmente como

$$\text{Si } \mathbf{A} \text{ entonces } \mathbf{B} = \mathbf{A} \Rightarrow \mathbf{B} \quad (3.27)$$

Donde  $\mathbf{A}$  y  $\mathbf{B}$  representan la precedencia y consecuencia, respectivamente. Sus dimensiones simbólicas vienen dadas por las de los vectores de estado y acción, respectivamente.

La lógica de toma de decisiones es la médula del dispositivo borroso. Ha de tener la capacidad de emular la toma de decisiones humana, basada en conceptos borrosos, y de inferir acciones borrosas empleando implicaciones borrosas y las reglas de inferencia de la lógica borrosa. Como apuntáramos, el producto Cartesiano es una de las operaciones más importantes en esa emulación. A partir de (3.27), la expresión del producto Cartesiano sería

$$\mathbf{A} \Rightarrow \mathbf{B} = \mathbf{A} \times \mathbf{B} = \mathbf{R} \quad (3.28)$$

Siguiendo una notación similar individualmente para cada implicación, tendríamos

$$R_i = A_i \times B_i; \quad i = 1, 2, \dots, p \quad (3.29)$$

donde,  $p$  es el número de implicaciones y  $R_i$  el conjunto borroso resultado del producto Cartesiano en la regla  $i$ -ésima.



A su vez, cada antecedente puede representarse como

$$A_i = A_i^1 \times A_i^2 \times \cdots \times A_i^n \quad (3.30)$$

Donde  $A_i^j$  es el conjunto borroso (uno de entre  $m_j$ ) de la variable de estado  $x_j$ ,  $j = 1, 2, \dots, n$  que participa en la regla  $i$ -ésima.

El producto Cartesiano se corresponde, en el antecedente, con el conectivo “Y”, que a su vez frecuentemente se hace corresponder con la intersección borrosa, de modo que

$$A_i = \bigcap_{j=1}^n A_i^j \quad (3.31)$$

Como se observa, hay dos facetas que pueden diferenciarse:

- El cálculo del producto Cartesiano  $A_i$  de los conjuntos borrosos del antecedente de la regla dada (3.28),
- El cálculo del producto Cartesiano del resultado anterior y el conjunto borroso de la consecuencia  $B_i$ , para la misma regla (3.27).

El primero tiene, para el DLB, una esencia estrictamente numérica. En cada instante, como resultado de la medición de las  $n$  variables de estado y su ulterior emborronado, se tiene, para cada regla, un conjunto de  $n$  valores de pertenencia  $\mu_{A_i^j}(x_j^0)$ ;  $i = 1, 2, \dots, p$ ;  $j = 1, 2, \dots, n$  ( $n$  “singletons”) y el producto Cartesiano se limita a encontrar otro valor de pertenencia  $\mu_{A_i^1 \times \cdots \times A_i^n}(x_1^0, \dots, x_n^0) = \alpha_i$ .

El segundo tiene otro cariz, pues, si bien el resultado de  $A_i = A_i^1 \times \cdots \times A_i^n$  es un número ( $\alpha_i \in [0, 1]$ ),  $B_i$  está caracterizado por una curva, la función de pertenencia  $\mu_{B_i}(u)$ , y  $R_i = A_i \times B_i$  es un conjunto borroso, caracterizado por otra función de pertenencia  $\mu_{R_i}$ , que depende de  $\alpha_i$  y de  $\mu_{B_i}$ .

Todas las  $p$  implicaciones (3.29) y de hecho cada consecuente  $B_i$ , son válidas en principio, sujetas, claro está, a la ponderación de su respectivo antecedente  $A_i$ . De modo que, la salida borrosa resultante saldrá de valorar de conjunto esos consecuentes ponderados, como se supone que lo haría el operador experto. El bloque de toma de decisiones suele simular esto mediante la unión:

$$R = \bigcup_{i=1}^p R_i \quad (3.32)$$

De donde resultarían los conjuntos borrosos correspondientes a las respectivas acciones de control. No obstante, la acción sobre el proceso exige que el DLB proporcione a su salida un valor determinado y no un conjunto borroso (por ejemplo: una velocidad de avance o una velocidad de giro dada), por lo que es necesario realizar una operación de *desemborronado*. El bloque de *desemborronado* sería entonces algo así como el interfaz de salida del DLB.

### 3.2.8 Características generales del DLB

Para analizar el DLB como un todo, se pueden aplicar dos filosofías: considerar incluidos o no, las interfaces de emborronado y *desemborronado* dentro de él como tal. En el primer caso estaríamos en presencia de un sistema determinista que podría ser descrito mediante:

$$\underline{u} = g(\underline{x}) \quad (3.33)$$

donde, para un cierto estado del sistema  $\underline{x}^o$ , se obtendría una acción de control  $\underline{u}^o$

$$\underline{u}^o = g(\underline{x}^o) \quad (3.34)$$

y  $g$  es una función vectorial no lineal. Esta no linealidad suele ser muy marcada y responde estrictamente a la presunta reproducción que hace el DLB de las habilidades del operador en la dirección del proceso, por tanto la linealización no sería conceptualmente adecuada, ni matemáticamente expedita. No obstante, no han faltado investigadores que la hayan intentado.

Si lo consideramos como parte del proceso, entonces el dispositivo es verdaderamente borroso (aplica conjuntos borrosos sobre conjuntos borrosos). El primer punto de vista subraya el carácter discreto del sistema y hace recordar a un relé de varios niveles. Mientras tanto el segundo criterio pudiera ser más rico y novedoso conceptualmente, pues refleja un dispositivo capaz de generar acciones cualitativas ante estados cualitativos.

Por tanto, de forma resumida el DLB se puede considerar como un sistema de aplicación de un espacio de entrada  $R_n$  a un espacio de salida  $Q_m$  con las siguientes propiedades:

- Biunívoco: considerando sólo dos variables de entrada al dispositivo, a la misma condición de entrada siempre le corresponde la misma condición de salida.
- Determinista: este mapa o función característica no cambia en el tiempo.
- No lineal: las variables de salida no son el resultado de una combinación lineal de la(s) variable(s) de entrada.

La Teoría del Control Moderna clasifica a estos sistemas como “Controladores Multivariantes No Lineales” o “Controladores de Múltiples Bandas”, y por tanto, algunos de los métodos de análisis de los sistemas de control no lineal podrían utilizarse en el DLB. A continuación se describen sus etapas internas.

### ***Emborronado***

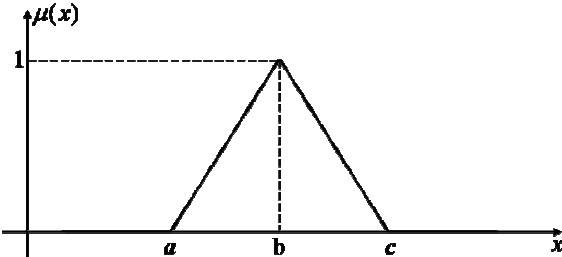
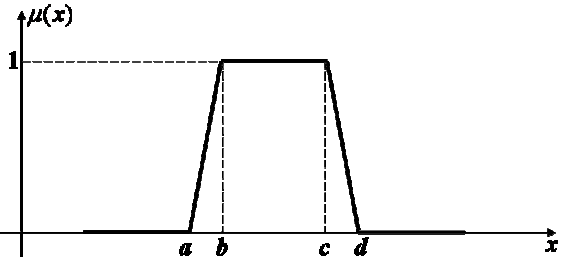
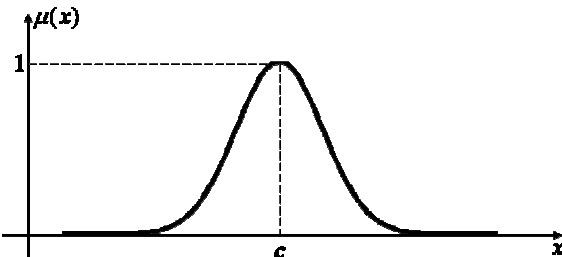
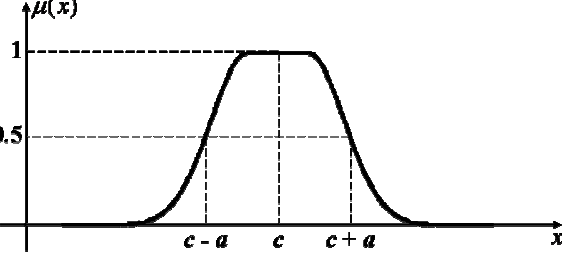
Este bloque ha de jugar el papel de interfaz entre las expresiones deterministas y las borrosas (cualitativas) de las variables de estado del proceso. Pensando en un operador humano, aunque que, si bien él interpreta cualitativamente el estado (la salida) del proceso, lo cierto es que objetivamente cada variable tiene, a cada instante, un valor (determinista) dado y tales valores son los que entran al DLB. De hecho, el emborronado sería la realización práctica de la definición de las variables como lingüísticas y de la partición de los universos de discurso: a cada valor (determinista) de la variable de estado, le corresponde cierto grado de pertenencia a los conjuntos borrosos de su respectivo universo, por lo que éstas se convierten en singularidades borrosas.

En la elección del número de particiones, o sea, el número de elementos del  $T(x)$  de la quintupla definida para la variable lingüística (ver Figura 3.6), influye decisivamente el método que se emplee para la creación del algoritmo del DLB. Uno de ellos, quizá el más popular, conocido como verbalización, es decir, la creación del algoritmo se basa exclusivamente en descripciones verbales de los operadores, acerca de su actuación en la dirección del proceso y resulta evidente que el número de conjuntos borrosos elegido para una variable tiene una esencia fuertemente subjetiva. No sería extraño que, para un mismo proceso, un operador considere que una variable lingüística dada tenga, digamos, 3 particiones cualitativas y otro considere que son 5 y como resultado, ambos dirigen "satisfactoriamente" el proceso.

Las funciones de pertenencia pueden ser expresadas de diferente forma. En la Tabla 3.2 se muestran algunas de las más usualmente utilizadas. Una excelente colección de estas funciones se ofrece en un trabajo de Kaufmann [180]. Como ya se ha mencionado, la elección del tipo de función es relativamente arbitraria y esta sujeta a factores tales como el tipo de DLB, los requisitos de cómputo, entre otros [187].

Se podría resumir entonces que el emborronado no es más que la transformación de la información determinista, enviada desde el entorno físico al DLB, en información cualitativa que usa como referencia a conjuntos borrosos. También se evidencia de la explicación, que esta transformación depende de parámetros del DLB fijados a priori, vinculados con la caracterización de las variables medidas del proceso como variables lingüísticas: número de conjuntos y forma de las funciones de pertenencia.

Tabla 3.2. Algunas funciones de pertenencia típicas.

FUNCIÓN	REPRESENTACIÓN GRÁFICA
<p>Triangular</p> $\mu(x) = \begin{cases} 0, & x \leq a; x \geq c \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \end{cases}$	
<p>Trapezoidal</p> $\mu(x) = \begin{cases} 0, & x \leq a; x \geq d \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \end{cases}$	
<p>Campana de Gauss</p> $\mu(x) = e^{\frac{-(x-c)^2}{2\sigma^2}}$	
<p>Campana generalizada (versión extendida de la función de distribución de Cauchy-Lorentz)</p> $\mu(x) = \frac{1}{1 + \left[ \left( \frac{x-c}{a} \right)^2 \right]^b}$	

### Base de Conocimiento

La base de conocimiento ha de incluir todos los datos e informaciones suministradas al DLB, tales que posibiliten el procesamiento matemático de las variables medidas del proceso, a fin de generar las salidas del DLB. Muchos autores sólo le adjudican las llamadas bases de reglas y bases de datos, ambas relacionadas con informaciones muy específicas del sistema experto borroso, como tal. Otros autores dan preferencia a completar esas informaciones con

otras propias de cualquier controlador [186]. Éstas abarcan los valores de referencia, valores límite de las variables, constantes varias de ajuste, parámetros necesarios para la *discretización* y normalización de las variables, entre otros. El número de niveles de cuantificación resulta de un compromiso entre la precisión y sensibilidad a que se aspira, y la memoria dinámica y velocidad de cómputo de que se dispone. Los intervalos más usuales para la normalización de los universos de discurso son  $[-1,1]$ ,  $[0,1]$ ,  $[-100\%,100\%]$  y  $[0,100\%]$ .

El conjunto de datos que conforman la base de datos incluye, en primer lugar, a todos aquellos imprescindibles para concretar la partición de todas las variables lingüísticas: Universos de discurso, número y nombre de las particiones así como la geometría de las funciones de pertenencia. Igualmente importantes son los relativos a las posibles alternativas para las operaciones y cálculos que han de realizarse en las etapas de toma de decisiones y *desemborronado*. Las características de la base de reglas determinan la necesidad de una atención particular a éstas.

### ***Base de reglas***

De todo lo comentado con anterioridad se deduce claramente la importancia de las *reglas* en su calidad de representación estructurada de “la política de manejo del operador”. La base de reglas comprende pues el conjunto de reglas de producción que deben ser prefijadas en el DLB. El número máximo posible de reglas viene dado por el producto de los números de particiones de todas las variables de entrada al DLB.

Para hablar de una base de reglas, es preciso ante todo elegir cuáles variables se utilizarán como de entrada y cuáles como de salida del DLB. No es ocioso subrayar que una selección adecuada de las variables de estado y acción del proceso resulta esencial para la caracterización y ulterior comportamiento del CLB. La experiencia y el conocimiento técnico desempeñan un papel importante en esta etapa de selección [186]. Más que otra cosa, se impone la necesidad de un análisis con enfoque sistémico, dónde emerjan no sólo las variables de entrada y salida, sino también los objetivos, restricciones y parámetros. Se pudiera decir que si esto es en general importante para cualquier sistema de control, lo es mucho más en el caso de los llamados procesos “complejos” (a los que preferentemente se aplica el control borroso) especialmente a la hora de concebir su base de reglas. Ejemplos típicos de variable de entrada de un DLB son el estado, el error del estado, la integral del error, etc.

Para crear las reglas que se prefijarán inicialmente en el DLB, existen tres vías que se clasifican como: *verbalización*, *emborronado*, e *identificación* [188].

A la primera ya se ha hecho referencia con anterioridad, al describir la etapa de emborronado del DLB. La segunda vía, denominada emborronado (no confundir con la etapa

de igual nombre del DLB) proporciona modelos borrosos a partir de expresiones matemáticas ordinarias por medio del principio de extensión de Zadeh. Tales expresiones matemáticas no suelen existir en los casos de procesos complejos. Por otra parte, la identificación supone la generación de descripciones causales, a partir de datos numéricos obtenidos midiendo las variables de estado y acción, durante la actuación de operadores eficientes.

Sobre la verbalización, la mayoría de los trabajos pioneros del control borroso de la década de los 70 se apoyaron en modelos basados en descripciones verbales de los operadores, tecnólogos, etc., básicamente a partir de dos métodos heurísticos: la interrogación del operador a partir de un cuestionario riguroso y/o una verbalización de la experiencia del operador. A pesar de estar afectadas por fuertes dosis de subjetivismo, constituye la técnica más popular.

La identificación, por su parte, provee la objetividad que emana de mediciones reales, procesando los archivos de datos de estas mediciones, para generar la base de conocimiento. Existen para ello diversas técnicas, entre las cuales está el emborronado de cada medición, convirtiéndola en un “número borroso” [189] y las técnicas de agrupamiento borroso (*fuzzy clustering*) que se han empleado para diseñar tanto controladores como modelos borrosos [190].

Takagi y Sugeno propusieron un interesante algoritmo que también se podría clasificar entre los de identificación, pues se basa en datos procedentes de mediciones sobre el proceso [24]. Las reglas que propusieron para su base de conocimiento eran de un nuevo tipo, con antecedente borroso pero consecuente determinista (algébrico):

$$SI \ x \ es \ A_i \ Y \dots Y \ y \ es \ B_i \ ENTONCES \ z = f(x, \dots, y) \quad (3.35)$$

donde  $x$  e  $y$  son variables lingüísticas,  $z$  es una variable ordinaria,  $A_i$  y  $B_i$  conjuntos borrosos y  $f$  una función ordinaria, frecuentemente lineal. El problema lo reducen a uno de estimación de parámetros por mínimos cuadrados. Los éxitos en los resultados de aplicaciones ulteriores de este algoritmo, incluido su perfeccionamiento, avalan el valor de este enfoque.

Por otra parte, Takagi y Sugeno ofrecen cuatro modos, no excluyentes, de obtención de las reglas: utilizando la experiencia del experto y conocimiento técnico del proceso, a partir de la experiencia de las acciones de control de los operadores, basada en modelos borrosos del proceso y basadas en el aprendizaje [191].

A continuación se mencionan algunas de las propiedades que deben cumplir la base de conocimientos [186, 192]:

- **Compleitud:** Aptitud de un algoritmo de control borroso de poder inferir una acción de control adecuada (no trivial) para cada estado del proceso. La integridad de un CLB depende de cuán completas sean tanto su base de datos como su base de reglas.

- **Continuidad:** Se dice que un conjunto de reglas es continuo si no contiene ninguna regla que produzca zonas “vacías” en el producto cartesiano. La integridad de la base de reglas es más fácil de garantizar introduciendo en el antecedente (o generando automáticamente) reglas que prevean todas las posibles combinaciones de atributos de las variables de entrada al controlador, a costa, claro está, del correspondiente gravamen de memoria.
- **Consistencia:** Las fuentes de obtención de las reglas pueden ser variadas, siendo los operadores humanos de las más importantes. Los errores subjetivos de éstos son potenciales fuentes de inconsistencia. El caso más claro está dentro de una base de reglas y sería el de reglas con igual (o casi igual) antecedente y muy diferente consecuente. Cuando el número de particiones de los universos de las variables de estado y acción es elevado, puede no resultar fácil, aún para el experto, el fijar un consecuente no inconsistente. En la literatura se reportan varias metodologías para los casos de variables de estado “clásicas” como el error, la derivada del error, etc. Una de ellas, se basa en justificar las reglas tomando como referencia una trayectoria de fase del sistema a lazo cerrado [193]. Otros autores proponen otro enfoque para sintetizar las reglas a partir de un modelo lingüístico de lazo cerrado, en lo que sería una versión borrosa del diseño por asignación de polos [194].
- **Interacción:** Un conjunto de reglas es interactivo o existe interacción, si el resultado que se obtiene de cada regla por separado es diferente a cuando se toma en cuenta el conjunto completo de reglas, lo que supone que el operador composicional utilizado modifica uno o más de los consecuentes.

### *Tipos de reglas*

Si bien es cierto que las más populares y conocidas reglas de un DLB son de la forma:

$$SI \begin{bmatrix} \text{variable de} \\ \text{estado 1} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix} Y \begin{bmatrix} \text{variable de} \\ \text{estado 2} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix} Y \dots ENTONCES \begin{bmatrix} \text{variable de} \\ \text{acción 1} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix} ; \begin{bmatrix} \text{variable de} \\ \text{acción 2} \end{bmatrix} ES \begin{bmatrix} \text{conjunto} \\ \text{borroso} \end{bmatrix} ; \dots \quad (3.36)$$

La lógica borrosa pone también a disposición del diseñador del sistema una muy rica variedad de tipos de reglas, lo que da una enorme flexibilidad en las opciones. A continuación se relacionan varios tipos recogidos en la literatura:

- Jerarquía de reglas complejas, empleadas en dispositivos borrosos expertos [195]. Las variables intermedias representan algún estado implicado (que no se puede medir) que puede afectar las acciones de control a aplicar:

$$\begin{array}{l}
SI \text{ [estado del proceso]} \text{ ENTONCES [variable intermedia 1]} \\
\vdots \\
SI \text{ [variable intermedia } N] \text{ ENTONCES [acción de control]}
\end{array} \quad (3.37)$$

- Metarreglas: dota a los dispositivos expertos borrosos de la inclusión de conocimientos de tipo estratégico. Las metarreglas permiten decidir cuáles reglas de bajo nivel han de aplicarse. Su uso es un importante recurso, al cual puede apelarse cuando se desee ahorrar volúmenes de cómputo y, con ello, aumentar las velocidades de generación de las salidas del DLB. Ejemplo de ellas:

$$\begin{array}{l}
SI \text{ [estado 1 del proceso]} \text{ ENTONCES [aplicar juego de reglas A]} \\
\vdots \\
SI \text{ [estado } N \text{ del proceso]} \text{ ENTONCES [aplicar juego de reglas B]}
\end{array} \quad (3.38)$$

- Secuencias de reglas de bajo nivel:

$$\begin{array}{l}
SI \text{ [estado del proceso]} \text{ ENTONCES [primero aplique el juego de reglas A]} \\
\text{[a continuación aplique el juego de reglas B]}
\end{array} \quad (3.39)$$

- De igual modo, existe la posibilidad de que las reglas sean función de objetivos y restricciones que pudieran usarse con propósitos de adaptación, de la forma

$$SI \text{ [nuevo objetivo]} \text{ ENTONCES [inicie grupo de reglas 1]} \quad (3.40)$$

Mención aparte merece la regla de “tipo Sugeno” mencionada con anterioridad (3.35), cuyo rasgo distintivo radica en el carácter determinista (función ordinaria) del consecuente y constituye uno de los primeros intentos, de reglas con consecuente no borroso. Vale la pena insistir finalmente, citando esta vez al propio Tong, que “la construcción de la base de reglas es el aspecto crucial y más difícil del diseño del CLB”.

### 3.2.9 La toma de decisiones

La toma de decisiones es la etapa que tiene la tarea de tratar de reproducir, lo más fielmente posible, los procesos intelectuales del operador humano en su accionar dirigiendo el proceso. De modo que todos los procesos matemáticos en ella implícitos, serán más adecuados cuanto mejor se simule la toma de decisiones humana.

La explicación realizada hasta ahora del DLB, en base de su esquema funcional (Figura 3.7) ha tenido un carácter esencialmente estático, pues se le ha concedido prioridad a la composición de la base de conocimiento. Seguidamente se dan a conocer las siguientes etapas de elaboración de las señales para la toma de decisiones. De utilizar una base de reglas de tipo clásico:



*SI < estado borroso del proceso > ENTONCES < acción de control borrosa >*

En cada instante  $t$ , el conjunto de reglas puede considerarse conceptualmente dividido en dos sub-conjuntos: uno de reglas activas (no nulas) y otro de reglas momentáneamente inactivas, pues ocurre que un estado dado  $x^o(t)$  dará lugar a la “activación” de un número de reglas. Serán activadas solamente aquellas reglas en las que todos los conjuntos borrosos del antecedente resulten ser no nulos. Algo similar habría que hacer si existe más de una variable de entrada. La desactivación puede realizarse indistintamente revisando regla a regla ó variable a variable. Este enfoque de la activación y desactivación resulta muy conveniente desde el punto de vista de las percepciones técnicas, como una fiel reproducción del más sano juicio de un operador experto.

Para el caso de un dispositivo múltiple entrada simple salida, el conjunto de reglas activas puede representarse de la siguiente forma:

$$\begin{aligned} &SI \ x_1 \text{ es } A_1^1 \ Y \ x_2 \text{ es } A_1^2 \ Y \dots Y \ x_n \text{ es } A_1^n \ ENTONCES \ u \text{ es } B_1 \\ &TAMBIÉN \\ &\vdots \\ &SI \ x_1 \text{ es } A_i^1 \ Y \ x_2 \text{ es } A_i^2 \ Y \dots Y \ x_n \text{ es } A_i^n \ ENTONCES \ u \text{ es } B_i \\ &TAMBIÉN \\ &\vdots \end{aligned}$$

donde cada atributo  $A_i^j$  ó  $B_i$  no son necesariamente diferente para cada regla. Los pasos siguientes se corresponden a operaciones lógicas borrosas. Pero antes es necesario tener en cuentas algunas consideraciones matemáticas acerca de la definición de los conectivos “Y” y “TAMBIÉN”.

En la mayoría de los DLB el conectivo “Y” se traduce prácticamente como una implicación borrosa en un espacio producto Cartesiano, en el que, las variables involucradas se desenvuelven en universos de discurso diferentes. Más generalmente, la implicación borrosa se asocia a las llamadas normas triangulares. Una norma triangular o *T-norma* es una función de dos lugares de  $[0,1] \times [0,1]$  en  $[0,1]$ , que debe cumplir los siguientes requisitos:

1.  $T(0,0) = 0$ ;  $T(a,1) = T(1,a) = a$
2. Si  $a \leq c$ ,  $b \leq d$  entonces  $T(a,b) \leq T(c,d)$  (monotonía)
3.  $T(a,b) = T(b,a)$  (conmutatividad)
4.  $T(T(a,b),c) = T(a,T(b,c))$  (asociatividad)

Además, debe satisfacer que

$$T_w(a, b) \leq T(a, b) \leq \min(a, b), \text{ siendo } T_w(a, b) = \begin{cases} a & \text{si } b = 1 \\ b & \text{si } a = 1 \\ 0 & \text{en otro caso} \end{cases}$$

Tales requisitos los cumplen, junto a las definiciones (3.22) y (3.23), con  $x_i \in [0, 1]$  para toda  $i$ :

$$\text{intersección} \quad x_i \cap x_j = \min\{x_i, x_j\} \quad (3.41)$$

$$\text{producto algébrico} \quad x_i \cdot x_j = x_i x_j \quad (3.42)$$

$$\text{producto acotado} \quad x_i \odot x_j = \max\{0, x_i + x_j - 1\} \quad (3.43)$$

$$\text{producto drástico} \quad x_i \cap x_j = \begin{cases} x_i & x_j = 1 \\ x_j & x_i = 1 \\ 0 & x_i, x_j < 1 \end{cases} \quad (3.44)$$

Las más utilizadas en el cálculo del producto Cartesiano de los DLB constituyen la intersección (Regla de implicación del mínimo de Mamdani) y el producto algébrico (Regla de implicación del producto de Larsen), que aparecen reflejadas en las ecuaciones (3.22) y (3.23).

Sobre el conectivo “*TAMBIÉN*” cabe apuntar, en primer lugar, que debe exhibir propiedades de conmutatividad y asociatividad, pues el orden de las reglas es irrelevante. Tales propiedades las poseen las antes mencionadas normas triangulares, aunque también las tienen las llamadas *co-normas triangulares* o *T-conorma*. Las *co-normas* son también funciones de dos lugares de  $[0, 1] \times [0, 1]$  en  $[0, 1]$  y las condiciones que deben cumplir son homólogas (o coincidentes) con las de las normas, a saber:

1.  $S(1, 1) = 1$ ;  $S(a, 0) = S(0, a) = a$
2. Si  $a \leq c$ ,  $b \leq d$  entonces  $S(a, b) \leq S(c, d)$  (monotonía)
3.  $S(a, b) = S(b, a)$  (conmutatividad)
4.  $S(S(a, b), c) = S(a, S(b, c))$  (asociatividad)

Además, debe satisfacer que:

$$S_w(a, b) \geq S(a, b) \geq \max(a, b), \text{ siendo } S_w(a, b) = \begin{cases} a & \text{si } b = 0 \\ b & \text{si } a = 0 \\ 1 & \text{en otro caso} \end{cases}$$

Esas condiciones las cumplen las siguientes definiciones, válidas para  $x_i \in [0, 1]$ , para toda  $i$ , como:

$$\text{unión} \quad x_i \cup x_j = \max\{x_i, x_j\} \quad (3.45)$$

$$\text{suma algébrica} \quad x_i \hat{+} x_j = x_i + x_j - x_i x_j \quad (3.46)$$

$$\text{suma acotada} \quad x_i \oplus x_j = \min\{1, x_i + x_j\} \quad (3.47)$$

$$\text{suma drástica} \quad x_i \cup x_j = \begin{cases} x_i & x_j = 0 \\ x_j & x_i = 0 \\ 1 & x_i, x_j > 0 \end{cases} \quad (3.48)$$

$$\text{suma disjunta} \quad x_i \Delta x_j = \max\{\min(x_i, 1 - x_j), \min(1 - x_i, x_j)\} \quad (3.49)$$

A mediados de la década de los 80 varios investigadores, especialmente polacos (Kiszka, Kochanska, Sliwiska, Stachowicz), trabajaron en la determinación de la definición más conveniente para los conectivos “Y” y “TAMBIÉN”, desde el punto de vista de la precisión y calidad del modelo borroso. De los resultados de éstos y otros investigadores e incluyendo consideraciones de orden práctico para la realización práctica del DLB se aconseja el uso de las Reglas de Mamdani ó de Larsen y el operador de unión para el conectivo *TAMBIÉN*. Esto se corresponde con los operadores “sup-min” de Zadeh y “sup-producto” de Kaufmann. De todos modos podría resultar prematuro el pretender llegar a conclusiones definitivas respecto a cuáles de los operadores son los mejores para los DLB. Ejemplo de ello es que Mizumoto reportó, ya en 1982, buenos resultados con sus operadores “sup-producto acotado” y “sup-producto drástico” [196].

### 3.2.10 Desemborronado

El bloque de *desemborronado* constituye la interfaz de salida del DLB, es el responsable de entregar al proceso físico, acciones deterministas a partir de conjuntos borrosos generados durante la toma de decisiones. La estrategia de *desemborronado* ha de tener como objetivo que las salidas deterministas resultantes sean las que “mejor” representen a esos conjuntos borrosos.

El procedimiento de *desemborronado* ha sido objeto de atención por parte de los investigadores, debido a que no existe un procedimiento sistemático para su realización, frecuentemente las estrategias de mejores cualidades para elegir la mejor  $u^\circ$  consumen elevados tiempos de cómputo, lo que va en detrimento de las prestaciones del DLB y la elección de una estrategia resulta de un compromiso entre virtudes y defectos de varias alternativas, entre otras razones.

Las estrategias de *desemborronado* más comúnmente usadas, desde los primeros DLB, son: el máximo, media del máximo (MDM) y el centro de área (CDA). La lógica de toma de decisiones genera una distribución posibilística  $\mu_R$  vs.  $u$  (ó un conjunto de distribuciones en el caso de varias salidas), de la que se debe extraer un valor determinista de  $u = u^\circ$ .

El criterio del máximo da lugar a salidas que corresponden a los valores de  $u$  en que las distribuciones alcancen su valor máximo, por vez primera en sentido de los valores de  $u$  crecientes.

$$u^\circ = \min \left( \arg \sup_{u \in U_u} (\mu_R(u)) \right) \quad (3.50)$$

La estrategia MDM genera como salida del DLB, los valores de  $u$  que representen la media de todos los valores que alcanzan el mismo máximo en la distribución posibilística resultante. Para el caso de universo discreto, la salida del DLB puede expresarse como:

$$u^\circ = \frac{1}{m} \sum_{u_j \in V} u_j \quad (3.51)$$

donde  $V$  representa un sub-conjunto ordinario de  $U$ , que agrupa los  $m$  puntos donde  $\mu_R(u) = \mu_R(u)_{\max}$  y  $m$  es la cardinalidad de  $V$ .

El método del centro de área (CDA), también conocido como del centroide, del centro de masa o de gravedad, se basa en calcular, como salida determinista, los valores de la variable  $u = u^\circ$  tal que divida en partes iguales el área bajo la curva de distribución posibilística resultante de la composición de las reglas. Analíticamente sería:

$$u^\circ = \frac{\int \mu_R(u) u du}{\int \mu_R(u) du} \quad (3.52)$$

En el caso de un universo discreto se calcula como:

$$u^{\circ} = \frac{\sum_j u_j \mu_R(u_j)}{\sum_j \mu_R(u_j)} \quad (3.53)$$

El método del CDA es el de mayor grado de coincidencia con el sentido común y la intuición de los especialistas a la hora de elegir la mejor  $u^{\circ}$ . La tendencia bastante frecuente, es de salidas más drásticas del MDM respecto al CDA y aún más drásticas para el máximo. Igualmente la composición “Sup-producto” da mayores valores de salida en igualdad de otras condiciones.

Desde inicios de los 80, varios investigadores estudiaron comparativamente estos tres métodos de *desemborronado* [197]. Es bastante generalizada la conclusión de que el método CDA ofrece los mejores resultados para determinar el mejor  $u^{\circ}$ . No obstante se plantea que, desde el punto de vista del comportamiento dinámico de un sistema de control a lazo cerrado, el método MDM ofrece un mejor comportamiento del sistema en régimen transitorio mientras que el CDA da mejor comportamiento a estado de régimen estacionario. Desde el punto de vista de la teoría de control clásica, empleando el método MDM, el DLB se comporta como un relé de varios niveles y con el CDA se obtienen resultados parecidos a los de un controlador proporcional integral (PI).

### 3.3 Estudio de los sistemas híbridos neuro-borrosos y su empleo en el control de sistemas complejos

A pesar de la amplia aplicación práctica de los sistemas de inferencia borrosa en los últimos años, existe aún la necesidad de disponer de métodos estándar que transformen, de forma eficiente y semiautomática, el conocimiento humano y la experiencia en base de reglas del DLB, además de métodos para el ajuste de los parámetros de las funciones de pertenencia de manera óptima.

Una vía para afrontar esta problemática es la utilización de técnicas de inteligencia artificial híbridas, basadas en sinergias y complementariedades. Dentro de los llamados sistemas inteligentes híbridos una de las estrategias de mayor éxito son los sistemas híbridos neuro-borrosos. Estos combinan el estilo de razonamiento humano de los sistemas borrosos con los algoritmos de aprendizaje y la estructura “conexionista” de las redes neuronales.

Debido a la creciente necesidad de desarrollar sistemas inteligentes adaptativos para el modelado y el control de procesos físicos, la combinación de las redes neuronales artificiales y los sistemas de inferencia borrosa ha atraído un interés creciente de los investigadores en diversas áreas científicas y de la ingeniería. Por una parte, las redes neuronales basan su aprendizaje ajustando las interconexiones entre sus capas, pero tiene la problemática de que el conocimiento aparece de forma implícita (valores numéricos sin un significado físico fácilmente reconocible) y suelen fallar cuando se producen comportamientos fuera de la

región de entrenamiento. Por otra parte, los sistemas de inferencia borrosa proporcionan un entorno o marco eficiente de computación basado en la imitación del razonamiento humano mediante el empleo de conjuntos borrosos y un modelo lingüístico constituido por un conjunto de reglas borrosas *SI-ENTONCES* (*IF-THEN*).

La hibridación de estos dos paradigmas permite aprovechar las ventajas de ambos, consiguiendo la sencillez de ajuste de las redes neuronales y a la vez disponiendo del conocimiento explícito expresado mediante el modelo lingüístico de los sistemas borrosos.

A partir de la segunda mitad de los años 80, aparecen en la literatura diversos trabajos en los cuales se introducen diferentes estructura de sistemas híbridos neuro-borrosos y que pueden ser separados en dos grandes grupos. Uno de ellos son las redes neuronales dotadas con la habilidad de manejar información borrosa para mejorar su capacidad de convergencia (*Fuzzy Neural Networks*, FNN) [198, 199]. El otro son los sistemas borrosos combinados con redes neuronales con el objeto de incorporar la capacidad de aprendizaje y así mejorar su precisión (*Neural-Fuzzy Systems*, NFS) [52, 200]. Por otra parte, en un trabajo anterior desarrollado por Farag et ál. se añade, además, las potencialidades de optimización de los algoritmos de computación evolutiva (por ejemplo, algoritmos genéticos) para el ajuste de las funciones de pertenencia de los sistema neuro-borrosos [78]. Una revisión en profundidad de las estrategias neuro-borrosas disponibles en la literatura, puede realizarse en los trabajos de Nauck et ál. [201] y de Li Xin Wang [202].

Referido a los sistemas borrosos que incorporan la capacidad de aprendizaje de las redes neuronales, en los últimos 15 años es posible encontrar, disponibles en la literatura, diversos algoritmos computacionales que emplean esta estrategia de hibridación para el ajuste de sistemas de inferencia borrosa. Entre ellos caben destacar los siguientes: ANFIS (*Adaptive Network based Fuzzy Inference System*) [52], FuNN (*Fuzzy Neural Network*) [203, 204], REFuNN (*Rules Extraction Fuzzy Neural Network*) [204], DENFIS (*Dynamic Evolving Neural-Fuzzy Inference System*) [205], HyFIS (*Hybrid neural Fuzzy Inference System*) [200], TWNFI (*Transductive Neuro-Fuzzy Inference model with Weighted data normalization*) [206] que facilitan tareas como el modelado y el control de sistemas no lineales y de gran complejidad [73, 207]. Es necesario destacar que varios de estos aportes han sido realizados por el grupo de investigación del Knowledge Engineering/Discovery Laboratory de la Universidad de Otago de Nueva Zelanda, liderado por el Dr. Nikola K. Kasabov.

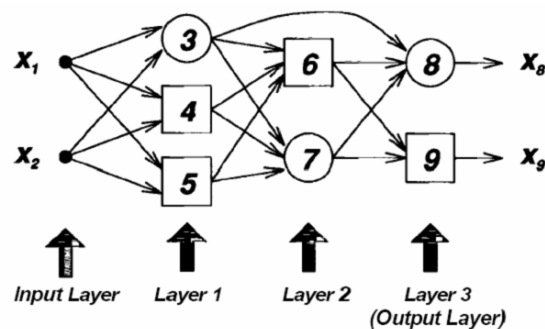
Entre los diferentes algoritmos computacionales neuro-borrosos mencionados, el conocido como *Adaptive Network based Fuzzy Inference System* (ANFIS) propuesto por Jyh-Shing R. Jang [52] es, además del trabajo pionero, uno de los más sencillos desde el punto de vista computacional y viable para aplicaciones en tiempo real.

ANFIS está basado en redes neuronales artificiales adaptativas, introducidas también por Jang [52] y las cuales no son más que redes neuronales artificiales multicapas con

alimentación hacia delante, diferenciándose en que las funciones de activación pueden ser distintas en cada nodo. La estructura y métodos de aprendizaje de la red neuronal, están diseñados de tal forma que puedan constituir un marco unificador de casi todos los diferentes paradigmas de redes neuronales artificiales con aprendizaje supervisado [208]. A continuación se realiza una breve introducción a la arquitectura de la red y sus procedimientos de aprendizaje.

### 3.3.1 Redes neuronales artificiales adaptativas

La estructura de la red adaptativa se compone de un conjunto de nodos conectados mediante enlaces direccionales. Cada nodo ejecuta una función estática a partir de sus entradas generando una salida única y cada enlace especifica la dirección del flujo de señal entre nodos. Las funciones de los nodos pueden ser parametrizables de tal forma que al cambiar los parámetros que definen la función, se cambia la función así como también todo el comportamiento de la red adaptativa. Los nodos realizan un mapeo estático de sus entradas a su salida, no existen estados dinámicos ni internos en cada nodo. Además, todas las funciones de nodos son diferenciables excepto en un número finito de puntos, lo cual facilita el desarrollo de algoritmos de aprendizaje.



**Figura 3.8. Representación en capas de la red neuronal adaptativa.**  
(Jang et ál., 1993 [208])

En general la red puede ser heterogénea, es decir, cada nodo puede tener una función de activación específica diferente a los otros. Los enlaces entre nodos solo especifican la dirección de propagación de las salidas de los nodos y por lo general no existen pesos u otros parámetros asociados, como en otros tipos de redes neuronales artificiales. La Figura 3.8 muestra una red adaptativa típica con dos entradas y dos salidas. Los nodos cuadrados representan nodos adaptativos mientras que los circulares son fijos, es decir, no tienen parámetros. El conjunto de parámetros de toda la red adaptativa es la unión del conjunto de parámetros de cada nodo adaptativo.

La red puede ser entrenada de forma híbrida, empleando el gradiente descendiente para ajustar los parámetros de las funciones de activación de los nodos y el método de mínimos cuadrados para ajustar nodos con funciones lineales. Además, el ajuste de los parámetros puede realizarse mediante entrenamiento por lotes (aprendizaje *off-line*), es decir, la actualización de los parámetros se realiza a partir de un conjunto de datos de entrenamiento. También el entrenamiento puede llevarse a cabo de forma incremental (aprendizaje *on-line*), es decir, los parámetros son actualizados para cada par de datos de entrada-salida.

A continuación se describe brevemente el método del gradiente descendiente que constituye la estrategia básica de aprendizaje. Se parte del supuesto de una red adaptativa de  $L$  capas, donde la capa  $k$  tiene  $N(k)$  nodos. El nodo en la posición  $i$  en la capa  $k$  se denota por  $(k, i)$ , su función de salida por  $f_i^k$  y su salida por  $x_i^k$ . Dado que la salida de cada nodo depende de sus entradas y de su conjunto de parámetros  $(\alpha, \beta, \gamma, \dots)$ , esta se expresa como

$$x_i^k = f_i^k \left( x_1^{k-1}, \dots, x_{N(k-1)}^{k-1}, \alpha, \beta, \gamma, \dots \right) \quad (3.54)$$

Asumiendo que el conjunto de datos de entrenamiento tiene  $P$  vectores, la medida del error o función de energía para el vector  $p$  ( $1 \leq p \leq P$ ) del conjunto de datos de entrenamiento, se calcula a partir de la suma de errores siguiente:

$$E_p = \sum_{m=1}^{N(L)} \left( d_{m,p} - x_{m,p}^L \right)^2 \quad (3.55)$$

donde  $d_{m,p}$  es la componente  $m$ -ésima del vector  $p$  y  $x_{m,p}^L$  es su correspondiente vector de salida para la evaluación actual del vector  $p$  en la red. Por tanto el error total y el cual es el objetivo a minimizar, se define de la siguiente forma:

$$E = \sum_{p=1}^P E_p \quad (3.56)$$

La definición del error en la ecuación (3.55) no es universal, para determinadas situaciones o aplicaciones el cálculo del error puede definirse de otra forma. Con el objetivo de desarrollar un procedimiento de aprendizaje que implementa un descenso de gradiente en  $E$  sobre el espacio de parámetros, primeramente se debe calcular la derivada del error para el vector de entrenamiento  $p$  y para cada nodo de salida  $x$ . A partir de la ecuación (3.55), el cambio del error a la salida del nodo  $i$  de la capa  $L$  se determina como:

$$\frac{\partial E_p}{\partial x_{i,p}^L} = -2 \left( d_{i,p} - x_{i,p}^L \right) \quad (3.57)$$



Para el nodo interno  $(k, i)$ , la derivada del error se calcula aplicando la regla de la cadena:

$$\frac{\partial E_p}{\partial x_{i,p}^k} = \sum_{m=1}^{N(k+1)} \frac{\partial E_p}{\partial x_{m,p}^{k+1}} \frac{\partial f_{m,p}^{k+1}}{\partial x_{i,p}^k} \quad (3.58)$$

siendo  $1 \leq k \leq L-1$ . La ecuación anterior se traduce en que la derivada del error o su tasa de cambio en los nodos internos, puede expresarse como una combinación lineal de las derivadas de los nodos en la capa siguiente. Por tanto, para calcular la tasa de cambio del error  $E_p$  con respecto a cada una de las salidas de los nodos que conforman toda la estructura de la red, primeramente se aplica la ecuación (3.57) para calcular la derivada del error en la capa de salida y a continuación se aplica la ecuación (3.58) de forma iterativa hasta la primera capa.

El vector gradiente se define como la derivada de la medida del error con respecto a cada parámetro. Si  $\alpha$  es un parámetro del nodo  $(k, i)$  y resulta que un conjunto de nodos  $S$  lo comparten, el gradiente del error respecto a él se determina aplicando también la regla de la cadena

$$\frac{\partial E_p}{\partial \alpha} = \sum_{x^* \in S} \frac{\partial E_p}{\partial x^*} \frac{\partial f^*}{\partial \alpha} \quad (3.59)$$

donde  $x^*$  y  $f^*$  constituyen la salida y la función, respectivamente, de un nodo genérico en  $S$ .

Por tanto, a partir de la ecuación anterior la derivada el error total  $E$  con respecto a  $\alpha$  se calcula como la suma

$$\frac{\partial E}{\partial \alpha} = \sum_{p=1}^P \frac{\partial E_p}{\partial \alpha} \quad (3.60)$$

Durante el aprendizaje el nuevo valor numérico del parámetro genérico  $\alpha$  se calcula sumando un incremento  $\Delta\alpha$  que, a partir del gradiente descendiente, puede expresarse de la siguiente forma

$$\Delta\alpha = -\eta \frac{\partial E}{\partial \alpha} \quad (3.61)$$

siendo  $\eta$  el factor de aprendizaje y según una longitud  $k$  (tamaño del paso) definida de cada transición del gradiente en el espacio del parámetro, puede determinarse así

$$\eta = \frac{k}{\sqrt{\sum_{\alpha} \left( \frac{\partial E}{\partial \alpha} \right)^2}} \quad (3.62)$$

El valor de  $k$  puede cambiarse para variar la velocidad de convergencia del algoritmo, pudiéndose utilizar para ello estrategias heurísticas.

Como se había comentado con anterioridad, el aprendizaje de la red puede realizarse mediante dos paradigmas. Por una parte, por medio del aprendizaje por lotes (*off-line*) donde la actualización de los parámetros se realiza a partir de un gran conjunto de datos de entrenamiento y para la actualización se utiliza la ecuación (3.60). Por otra parte, a través de un aprendizaje incremental (*on-line*) donde los parámetros pueden ser actualizados mediante un par de datos entrada-salida, empleando en este caso la ecuación (3.59). Esta última estrategia resulta de gran utilidad para la identificación, en tiempo real, de sistemas dinámicos con parámetros variables en el tiempo.

### ***Aprendizaje híbrido.***

El proceso de identificación del conjunto de parámetros de la red a través del gradiente descendiente, en ocasiones, resulta lento hasta que logra una convergencia y puede quedar atrapado en mínimos locales. En el caso de que alguna salida de la red neuronal sea lineal con respecto a algunos de los parámetros, el algoritmo de aprendizaje puede combinarse con el método de mínimos cuadrados y de esta forma acelerar el proceso de identificación de parámetros.

Por ejemplo, si se asume una red adaptativa con una sola salida, a partir de un vector de entrada a la red  $\mathbf{i}$  y el conjunto de parámetros  $S$ , la salida puede representarse de esta forma

$$o = F(\mathbf{i}, S) \quad (3.63)$$

siendo  $F$  la función que implementa la red.

De existir una función  $H$  tal que la función compuesta  $H \circ F$  es lineal en algunos elementos de  $S$ , este conjunto puede quedar dividido en dos conjuntos  $S_1$  y  $S_2$ , representándose como la suma de estos, es decir,  $S = S_1 \oplus S_2$ . De esta forma, el conjunto  $S_2$  puede contener los elementos para los cuales  $H \circ F$  es lineal. Aplicando  $H$  a la ecuación de salida de la red se tiene

$$H(o) = H \circ F(\mathbf{i}, S) \quad (3.64)$$

La ecuación anterior es lineal respecto a  $S_2$ , por lo que si el conjunto  $S_1$  se determina previamente, entonces al evaluar los datos de entrenamiento en la ecuación anterior se obtiene la siguiente ecuación matricial

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{B} \quad (3.65)$$

donde  $\boldsymbol{\theta}$  es el vector de parámetros desconocidos que pertenecen a  $S_2$ . Esta ecuación matricial constituye un problema estándar de regresión lineal, pudiéndose resolver empleando mínimos cuadrados.

En el caso del aprendizaje por lotes, se utilizarían los  $P$  vectores de entrenamiento. Pero para el caso del aprendizaje incremental, es necesario utilizar, además del par de datos entrada-salida actual, datos de entrenamiento de instantes de tiempo anteriores, pero introduciendo un factor de olvido, de tal forma que para la obtención de los parámetros  $\boldsymbol{\theta}$ , se atenúe la influencia de los datos de entrenamiento de instantes anteriores.

La actualización del conjunto total de parámetros de la red puede realizarse combinando el método del gradiente descendiente y mínimos cuadrados. Cada iteración del procedimiento de aprendizaje híbrido se compone de un paso hacia delante y uno hacia atrás. En el paso hacia delante, se introducen los datos de entrada y se calculan cada una de las salidas de los nodos, capa por capa de la red, hasta completar las matrices  $\mathbf{A}$  y  $\mathbf{B}$  de la ecuación matricial (3.65). A partir de estas matrices, empleando mínimos cuadrados se obtienen los elementos de  $S_2$  y se calcula el error  $E_p$  para cada par de datos entrada-salida de entrenamiento. En el paso hacia atrás, las derivadas de los errores se propagan desde la salida hacia la entrada, actualizándose los parámetros de  $S_1$  mediante el método del gradiente.

Este procedimiento de aprendizaje híbrido reduce el espacio de búsqueda para el método de gradiente conllevando, por tanto, una disminución importante del tiempo de convergencia del algoritmo de aprendizaje.

Las redes neuronales artificiales adaptativas pueden ser aplicadas en diversas áreas. Para un estudio en profundidad de los métodos de aprendizaje, así como de ejemplos de aplicaciones, se puede consultar los trabajos de Jang [52], Jang et ál. [53, 208], Denai et ál. [73], entre otros.

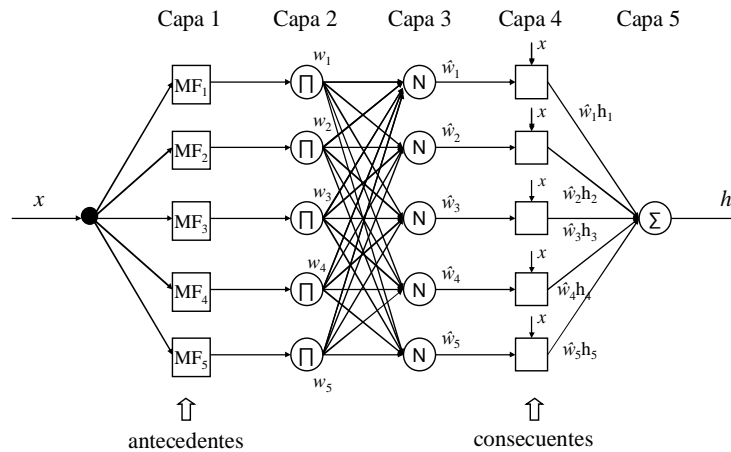
En la siguiente sección se introduce una estructura de red equivalente a sistemas de inferencia borrosa. Esta arquitectura de red se ha denominado ANFIS, que no es más que un sistema de inferencia borrosa representado por una red neuronal adaptativa [52]. Ha continuación se describe ANFIS para sistemas de inferencia de tipo Takagi-Sugeno-Kang, donde la base de reglas determina la topología de la red y representando los nodos en

determinadas capas de la red, los parámetros de las funciones de pertenencia de las entradas y los coeficientes de los polinomios de salida de cada regla.

### 3.3.2 Arquitectura ANFIS

El principio de funcionamiento del algoritmo ANFIS se basa en la extracción de reglas borrosas en cada nivel de una red neuronal. Una vez obtenidas las reglas, éstas deben proporcionar la información necesaria del comportamiento global del proceso.

ANFIS implementa, principalmente, el modelo de Takagi-Sugeno-Kang para la estructura de base de reglas (3.35) del sistema borroso, aunque los sistemas de inferencia de tipo Mamdani también pueden ser representados mediante redes adaptativas. La arquitectura de ANFIS dispone de cinco capas tal y como se muestra en la Figura 3.9, donde se ilustra a modo de ejemplo una estructura de una entrada y una salida. Los nodos representados con cuadrados son nodos cuyos parámetros son ajustables, mientras que los nodos representados por círculos son nodos fijos.



**Figura 3.9. Arquitectura del sistema ANFIS.**

A continuación se presenta ANFIS para el caso particular de un sistema de una entrada y una salida como el de la Figura 3.9 y 5 nodos en las capas ocultas. En la primera capa se produce el emborronado (3.66). La salida de cada nodo se representa por  $O_{l,i}$ , donde  $i$  es el  $i$ -ésimo nodo de la capa  $l$ .

$$O_{1,i} = \mu_{A_i}(x) \quad (3.66)$$

siendo  $x$  la variable de entrada al nodo y  $A_i$  es el conjunto borroso asociado al nodo. Si se utiliza una versión extendida de la función de distribución de Cauchy-Lorentz, también denominada campana generalizada (ver Figura 3.10), como función de pertenencia borrosa,

obtendríamos la siguiente expresión donde  $a_i$ ,  $b_i$  y  $d_i$  son los parámetros antecedentes ajustables o “parámetros premisas”:

$$\mu_{A_i}(x) = \frac{1}{1 + [((x - d_i) / a_i)^2]^{b_i}} \quad (3.67)$$

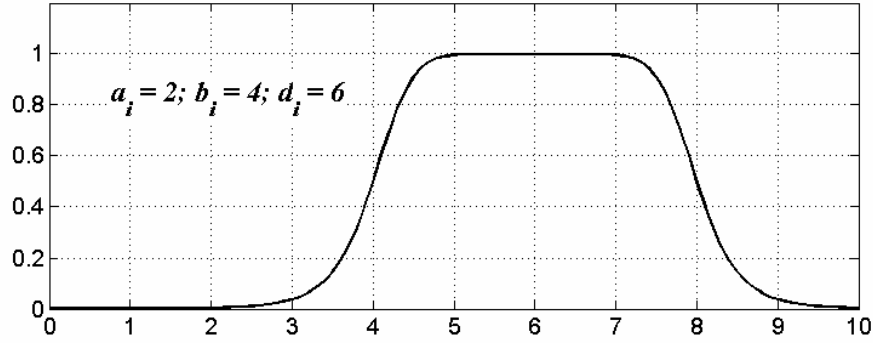


Figura 3.10. Ejemplo de función de distribución de tipo campana generalizada.

De hecho, además de esta función de pertenencia, cualquier función continua o diferenciable por tramos, tales como las funciones de pertenencia de tipo trapezoidal o triangular, puede ser utilizada en esta capa.

En la segunda capa se multiplican las señales de entrada. La salida de esta capa es el resultado de aplicar la regla del máximo (varios operadores *T-normas* pueden ser utilizados como función del nodo). La tercera capa normaliza la importancia de cada regla.

$$O_{2,i} = w_i = \prod \mu_{A_i}(x) \quad (3.68)$$

$$O_{3,i} = \bar{w}_i = \frac{w_i}{\sum_i w_i}, \quad i = 1, 2, \dots, 5 \quad (3.69)$$

La cuarta capa calcula el consecuente, es decir, la función polinómica de primer orden para cada regla borrosa, donde  $m_i$  y  $c_i$  son los “parámetros consecuentes”.

$$O_{4,i} = \bar{w}_i h_i = \bar{w}_i (m_i x + c_i) \quad (3.70)$$

Por último, la quinta capa realiza el *desemborronado* como sigue:

$$O_{5,1} = h = \sum_i \bar{w}_i h_i = \frac{\sum_i w_i h_i}{\sum_i w_i}, \quad i = 1, 2, \dots, 5 \quad (3.71)$$

Como puede observarse en la arquitectura ANFIS, dados los parámetros premisas, a partir de las ecuaciones (3.71) y (3.70) la salida final del sistema puede expresarse como una combinación lineal de los parámetros consecuentes

$$h = \sum_i \bar{w}_i (m_i x + c_i) = \sum_i (\bar{w}_i x) m_i + \sum_i (\bar{w}_i) c_i \quad (3.72)$$

Por tanto el conjunto total de parámetros  $S$  de la arquitectura, puede dividirse en los conjuntos de parámetros premisas ( $S_1$ ) y de parámetros consecuentes ( $S_2$ ), por lo que el procedimiento de aprendizaje híbrido de las redes adaptativas puede aplicarse directamente a la estructura ANFIS.

En el paso hacia delante, se calculan las salidas de cada nodo hasta la capa 4 y el consecuente se calcula mediante mínimos cuadrados. Cuando los valores de los antecedentes son fijos, la salida general del sistema puede expresarse como una combinación lineal de los consecuentes (3.72).

$$h = \begin{bmatrix} \bar{w}_1 x & \bar{w}_1 & \cdots & \bar{w}_5 x & \bar{w}_5 \end{bmatrix} \cdot \begin{bmatrix} m_1 \\ c_1 \\ \vdots \\ m_5 \\ c_5 \end{bmatrix} = \mathbf{A} \cdot \boldsymbol{\theta} \quad (3.73)$$

A partir de la salida deseada  $h$  y la matriz  $\mathbf{A}$ , se aplica el método de mínimos cuadrados para determinar el vector  $\boldsymbol{\theta}$ .

En el segundo paso o paso hacia atrás, los errores son propagados hacia atrás para modificar los parámetros premisas (ej.,  $a_i$ ,  $b_i$  y  $d_i$ ) mediante el método de gradiente descendiente. Por ejemplo, para el caso de los parámetros  $a_i$  de las funciones de pertenencia, en cada iteración  $n$  y dada la tasa de aprendizaje  $\eta$  estos se modifican como

$$a_i(n+1) = a_i(n) - \eta \frac{\partial E}{\partial a_i} \quad (3.74)$$

El procedimiento de aprendizaje se repite hasta que se alcanza alguna condición de parada o criterio de error.

La utilización de sistemas de inferencia borrosa representados mediante redes adaptativas en sistemas de control, tuvo sus primeras realizaciones a mediados de los años 90, donde varios investigadores desarrollaron aplicaciones para el control de variables de procesos continuos [209-211]. A día de hoy, son innumerables las aplicaciones de ANFIS a la monitorización y/o control de procesos en multitud de esquemas [212, 213].

### 3.4 Conclusiones

En este capítulo se ha realizado una revisión de tres técnicas que por su consolidado cuerpo de doctrinas e impacto científico y técnico se han considerado en esta Tesis Doctoral con vistas a la regulación automática de procesos complejos. Se han expuesto las ideas conceptuales y básicas de estrategias de control clásicas lineales, como el algoritmo de control proporcional, integral y derivativo, PID, de muy amplia utilización y basadas en *soft-computing*, como es el caso de los sistemas de inferencia borrosa. Por otra parte, se ha abordado el estudio de estrategias de hibridación de los sistemas borrosos con otras técnicas de inteligencia artificial, como las redes neuronales artificiales, combinando de esta forma el estilo de razonamiento humano de los sistemas borrosos con la estructura conexionista y los algoritmos de aprendizaje de las redes neuronales. Esto implicará, por tanto, una mejora del desempeño de los sistemas borrosos en la monitorización y control de sistemas complejos.

Los algoritmos de control estudiados serán examinados en el Capítulo 5 para el diseño de sistemas de control de procesos complejos. Se examinará, la potencialidad de los sistemas híbridos neuro-borrosos y del control clásico PID para el control de un proceso de fabricación. Además, como parte de la metodología de diseño de sistemas de control, se comprobará también el desempeño de estrategias de hibridación de la lógica borrosa y diferentes métodos metaheurísticos, para el ajuste óptimo del DLB como estrategia de control de procesos complejos. En específico se explorarán las sinergias con los algoritmos genéticos, el algoritmo de temple simulado, el algoritmo de Nelder-Mead y el algoritmo de entropía cruzada, de los cuales se realizará una introducción en el siguiente capítulo.





---

## **CAPÍTULO 4**

# **ESTUDIO DE MÉTODOS DE OPTIMIZACIÓN DE INSPIRACIÓN GEOMÉTRICA, FÍSICA Y BIOLÓGICA EN EL CONTROL DE PROCESOS COMPLEJOS**

---

Las estrategias de control introducidas en el capítulo anterior, tanto basadas en *soft-computing* como lineales, serán empleadas para el diseño de sistemas *inteligentes* de regulación automática de procesos no lineales y con relativa alta complejidad. Con vistas a desarrollar procedimientos de ajuste óptimo de estas estrategias, se hace necesario llevar a cabo un estudio de métodos de optimización de diversas fuentes de inspiración.

Los algoritmos que serán considerados en esta Tesis Doctoral (Nelder-Mead, temple simulado, algoritmos genéticos y entropía cruzada) han sido seleccionados, principalmente, por tener varios elementos en común. Comparten el hecho de no depender del gradiente y de la Hessiana, tienen un basamento matemático teórico y práctico bien fundamentado y una gran aceptación por la comunidad científica. Además, son bien conocidos sus buenos resultados en la obtención o aproximación de mínimos globales de funciones inherentemente no-lineales.

En las siguientes 5 secciones del capítulo se introducen los conceptos básicos de los algoritmos de optimización, así como una estrategia para su aplicación al ajuste de sistemas de control de procesos complejos (sección 4.1). En la sección 4.2 se introduce el método de optimización Nelder-Mead [21, 214] que es una técnica de optimización no-lineal que goza de gran popularidad y además no utiliza información sobre el gradiente de los datos. Esto hace que resulte adecuado para ser aplicado en problemas con funciones discontinuas o con derivadas abruptas.

La sección 4.3 muestra un estudio del algoritmo de optimización temple simulado o recocido simulado (*Simulated Annealing*) [17, 215], que es un método de búsqueda meta-heurística para la resolución de problemas de optimización global y está inspirado en el proceso físico de recocido de metales.

A continuación, en la sección 4.4, se hace una breve introducción al estudio de algoritmos genéticos [36, 216, 217] que es un método, también muy maduro científicamente basado en la selección natural y ha sido, además, ampliamente utilizado en la solución de diversos tipos de problemas de optimización como, por ejemplo, el ajuste óptimo de sistemas de control basados en técnicas de *soft-computing* para la regulación automática de procesos complejos. Adicionalmente, en la sección 4.5 se exponen las ideas conceptuales de uno de los métodos de optimización global más novedosos: el algoritmo de entropía cruzada (*Cross Entropy*) [20, 218, 219]. Este algoritmo se enmarca dentro de los métodos de tipo Monte Carlo y permite la optimización de problemas combinatoriales, continuos y con multiplicidad de extremos.

#### 4.1 Aplicación de algoritmos de optimización al ajuste de sistemas de control de procesos complejos

El empleo de estrategias de optimización como las que se estudiarán en este capítulo, puede resultar de gran utilidad. Además, desde el punto vista computacional, pueden ser eficientes para el ajuste óptimo de los parámetros de los sistemas de control de procesos complejos.

Consideremos un proceso simple entrada-simple salida gobernado por un sistema de control como el mostrado en la Figura 4.1, donde  $F$  es la variable controlada del proceso,  $\Delta f$  la variable de entrada del proceso y acción de control,  $e$  es la variable de error entre la variable de referencia  $F_R$  del controlador y la salida del proceso y  $F_Z$  es una perturbación a la salida del proceso.

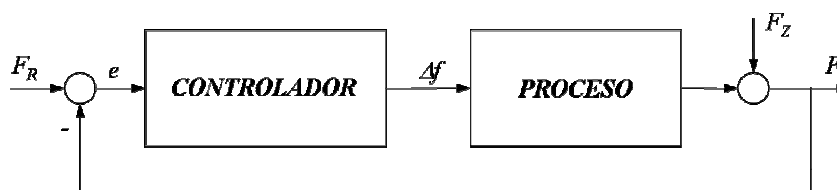


Figura 4.1. Diagrama general de un sistema de control de un proceso físico.

Consideremos que el controlador se define mediante una ley de control que relaciona su variable de entrada (el error  $e$ ) con la acción de control  $\Delta f$  y, siendo  $\mathbf{v}$  un vector de parámetros, ésta se puede expresar en el dominio temporal  $t$  de la siguiente forma:

$$\Delta f(t) = c(e(t), \mathbf{v}); e(t) = F_R(t) - F(t) \quad (4.1)$$

Por lo general, la ley de control se diseña persiguiendo ciertos objetivos (objetivos de control) como pueden ser, por ejemplo, que durante un intervalo de tiempo definido de la respuesta temporal, la salida del proceso  $F$  tenga muy poca variación respecto a la referencia, es decir,  $e \approx 0$  o muy pequeño en todo el intervalo. Otro objetivo de control puede ser que la salida  $F$  cumpla algún modo de respuesta determinado como, por ejemplo, que no oscile alrededor de  $F_R$  que se traduce también en oscilaciones del error  $e$  alrededor de 0 o la amplitud de las posibles oscilaciones sean mínimas. También, otro posible objetivo de control puede ser que el valor absoluto del error ( $|e|$ ) alcance y se mantenga, además, por debajo de cierto valor  $e_{\max} > 0$ , durante un intervalo de tiempo de establecimiento  $T_S$ , entre varios otros objetivos posibles.

Definamos cierto intervalo de tiempo  $[t_0, t_F]$  correspondiente a la respuesta temporal del sistema de control, ante cambios en la perturbación  $F_Z$  o su referencia  $F_R$ . A partir de la respuesta temporal del error  $e(t)$  es posible definir, en el intervalo  $[t_0, t_F]$ , cierta función de coste  $g$  que evalúe el comportamiento dinámico del sistema de control mediante una cifra de mérito  $I$ .

$$I = g(e, t) \Big|_{t_0}^{t_F} \quad (4.2)$$

El objetivo de control que se plantee puede cuantificarse mediante esta cifra de mérito. Existen varias cifras de mérito o funciones de coste de un amplio uso en el diseño de sistemas de control de procesos físicos, que son empleadas tanto en su etapa de diseño como de evaluación. Entre las funciones de coste, más ampliamente utilizadas, se encuentra la integral del error absoluto ponderado en el tiempo (*ITAE*) (4.3), que permite obtener una respuesta transitoria del sistema de control con un pequeño tiempo de establecimiento y sin oscilaciones.

Otros criterios también definidos son la integral del error cuadrático ponderado en el tiempo (*ITSE*) (4.4), la integral del error absoluto (*IAE*) (4.5) y la integral del error cuadrático ponderado por el tiempo al cuadrado (*IT<sup>2</sup>SE* o *ISTSE*) (4.6), entre otros. En específico, el criterio *ITAE* penaliza las respuestas transitorias de larga duración y es muy sensible ante variaciones de parámetros de los sistemas de control en comparación con otros criterios. Por otra parte, el diseño basado en el criterio *ITAE*, por lo general, causa menores oscilaciones en la variable controlada del proceso que los criterios *IAE* e *ITSE*.

$$ITAE = \int_{t_0}^{t_F} |e(t)| \cdot t \, dt \quad (4.3)$$

$$ITSE = \int_{t_0}^{t_F} e^2(t) \cdot t \, dt \quad (4.4)$$

$$IAE = \int_{t_0}^{t_F} |e(t)| \, dt \quad (4.5)$$

$$ISTSE = \int_{t_0}^{t_F} e^2(t) \cdot t^2 \, dt \quad (4.6)$$

Una vez definida una cifra de mérito, el procedimiento de optimización puede realizarse, tanto en tiempo real (*on-line*), es decir, empleando datos reales de  $F_R$ ,  $F$  y  $\Delta f$  como en tiempo de simulación (*off-line*) utilizando algún modelo del proceso físico que relacione su entrada y su salida:

$$F(t) = G(\Delta f, t) \quad (4.7)$$

Consideremos que el procedimiento de optimización se realiza *off-line* empleando, por tanto, el modelo del proceso. El cálculo de la cifra de mérito  $I$  se realiza como:

$$\begin{aligned} I &= g(e, t) \Big|_{t_0}^{t_F} = g(F_R(t) - F(t)) \Big|_{t_0}^{t_F} \\ &= g(F_R(t) - G(\Delta f, t)) \Big|_{t_0}^{t_F} \\ &= g(F_R(t) - G(c(e(t), \mathbf{v}))) \Big|_{t_0}^{t_F} \end{aligned} \quad (4.8)$$

Empleando la ecuación anterior como función objetivo, es posible minimizar la cifra de mérito  $I$ , ajustando el vector de parámetros  $\mathbf{v}$  de la ley de control como entrada a la función. Por ejemplo, si se utiliza la cifra de mérito  $ITAE$  la función de coste se expresa de la siguiente forma:

$$ITAE = \int_{t_0}^{t_F} |F_R(t) - G(c(e(t), \mathbf{v}))| \cdot t \, dt \quad (4.9)$$

En la Figura 4.2 puede observarse un diagrama de bloques del procedimiento de ajuste óptimo del sistema de control que ha sido muy frecuentemente empleado, tanto en la literatura como en procesos físicos, ya sea en tiempo real o en simulación.

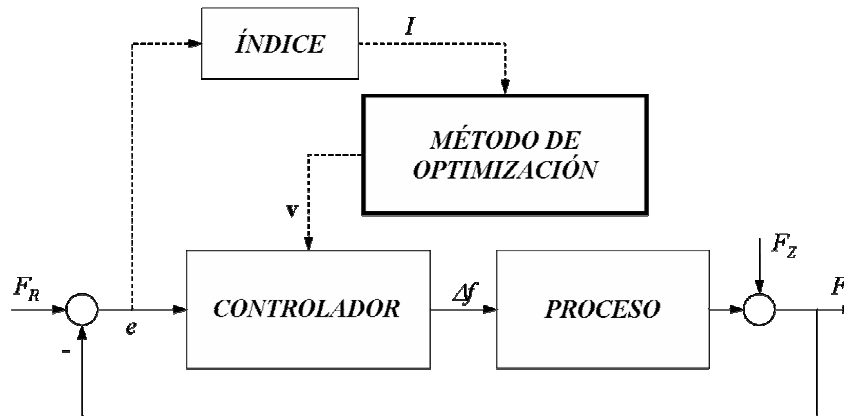


Figura 4.2. Diagrama de bloques del sistema de control y algoritmo de optimización.

## 4.2 Estudio del método de optimización Nelder-Mead para su aplicabilidad al ajuste de sistemas de control de procesos complejos

El algoritmo de Nelder-Mead, publicado en 1965 por Nelder y Mead [21], es un método numérico para minimizar una función objetivo en un espacio multidimensional. La aplicación del algoritmo básico es relativamente fácil, además de su comprensión, por lo que su utilización ha resultado muy popular en muchos campos de la ciencia y la tecnología. El método no utiliza información sobre el gradiente de los datos, por lo que resulta adecuado para ser aplicado en problemas con funciones discontinuas o con derivadas abruptas [214] y se enmarca dentro de la clase de métodos de búsqueda directa [220].

En el método se utiliza el concepto de un *símplex*, que es una forma geométrica de envolvente convexa y  $N+1$  vértices, correspondiéndose  $N$  con el número de parámetros a optimizar. El paso básico del método consiste en sustituir el peor punto del *símplex* con otro punto reflejado en el resto de puntos [14].

Si se considera una función no lineal  $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$ , el método comienza con un conjunto de  $n+1$  puntos  $x_0, \dots, x_n \in \mathfrak{R}^n$ , considerados los vértices de un *símplex*  $S$  y el correspondiente valor de la función  $f$  evaluada en los puntos vértices ( $f_j = f(x_j)$ ,  $j = 0, \dots, n$ ).

A partir de estas condiciones iniciales, el método realiza una secuencia de transformaciones en el *símplex*  $S$  definido, encaminadas en reducir el valor que toma la función evaluada para cada vértice. En cada paso, la transformación está determinada por el cálculo de uno o más puntos de prueba, junto al valor correspondiente de la función para estos puntos, y se comparan estos valores de la función con los correspondientes a los vértices. El proceso culmina cuando el *símplex*  $S$  es lo suficientemente pequeño según cierto criterio de evaluación.

En cada iteración del método de Nelder-Mead (NM), inicialmente se determinan para el actual *símplex*  $S$  los índices  $h$ ,  $s$  y  $l$ , correspondientes al peor, segundo peor y el mejor vértice, respectivamente

$$f_h = \max_j f_j, f_s = \max_{j \neq h} f_j, f_l = \min_j f_j \quad (4.10)$$

Seguidamente se calcula el centroide  $x_c$  de todos los puntos, excepto el punto  $x_h$ , de la forma

$$x_c = \frac{1}{n} \sum_{j \neq h} x_j \quad (4.11)$$

A continuación se construye un nuevo *símplex* a partir del actual. Para ello se reemplaza solamente el peor vértice  $x_h$  por un mejor punto, por medio de transformaciones de reflexión, expansión o contracción con respecto al lado opuesto a  $x_h$ .

Las coordenadas del punto de reflexión  $x_r$  de  $x_h$  se calculan mediante la relación

$$x_r = x_c + \alpha(x_c - x_h) \quad (4.12)$$

siendo  $\alpha$  una constante positiva y denominado coeficiente de reflexión. Por tanto,  $x_r$  se encuentra en la línea definida entre los puntos  $x_h$  y  $x_c$ , en el lado opuesto a  $x_h$ . Definiendo  $d_{ij}$  como la distancia entre dos puntos  $x_i$  y  $x_j$  cualesquiera, entonces  $\alpha$  relaciona la distancia entre los puntos  $x_r$ ,  $x_c$  y entre los puntos  $x_h$ ,  $x_c$ , es decir:

$$d_{rc} = \alpha d_{hc} \quad (4.13)$$

Si la función evaluada en el punto de reflexión,  $f_r = f(x_r)$ , se encuentra entre  $f_h$  y  $f_l$  ( $f_l \leq f_r < f_s$ ), entonces el vértice  $x_h$  se reemplaza por  $x_r$  reiniciando el proceso con el nuevo *símplex*.

En caso contrario, si  $f_r < f_l$  significa que la reflexión ha producido un nuevo mínimo y por tanto se expande el punto  $x_h$  a un punto  $x_e$  determinado como:

$$x_e = x_c + \gamma(x_r - x_c) \quad (4.14)$$

El parámetro  $\gamma$  se denomina coeficiente de expansión, siendo mayor que la unidad, y constituye la razón entre las distancias  $d_{ec}$  y  $d_{rc}$  ( $\gamma = d_{ec} / d_{rc}$ ). Existen dos enfoques para el

reemplazo de  $x_h$ , uno de ellos, que es el utilizado en el algoritmo original propuesto por Nelder y Mead [21], si  $f_e < f_l$  entonces se reemplaza el punto  $x_h$  por  $x_e$  y se reinicia el proceso con el nuevo símplex sin importar la relación entre  $f_r$  y  $f_e$ . Pero puede suceder que  $f_r < f_e$  y por tanto  $x_r$  es mejor punto que  $x_e$ . Otro enfoque propuesto por Lagarias et ál. [221], que es el más empleado en las aplicaciones del algoritmo, considera como nuevo vértice el mejor de los puntos  $x_r$  y  $x_e$ , produciéndose solamente la expansión si  $f_e < f_r < f_l$ .

Sin embargo, si resultase  $f_r \geq f_s$  entonces, a partir de un coeficiente de contracción  $\beta$ , se calcula un punto de contracción  $x_t$  empleando el mejor de los puntos entre  $x_h$  y  $x_r$ , teniendo dos casos posibles:

- Si  $f_s \leq f_r < f_h$ , es una contracción hacia afuera y el punto de contracción se calcula como:

$$x_t = x_c + \beta(x_r - x_c), \quad \beta = d_{tc} / d_{rc} \quad (4.15)$$

- Si  $f_r \geq f_h$ , se produce entonces una contracción hacia adentro y el punto de contracción se calcula como:

$$x_t = x_c + \beta(x_h - x_c), \quad \beta = d_{tc} / d_{hc} \quad (4.16)$$

Al evaluar la función  $f$  para el punto de contracción  $x_t$  ( $f_t = f(x_t)$ ), si en el primero de los casos anteriores resulta que  $f_t \leq f_r$ , entonces se acepta  $x_t$  y se reinicia el proceso. Este hecho también se produce para el segundo caso (contracción hacia dentro) si se confirma que  $f_t < f_h$ . Si no se cumple ninguna de las condiciones para  $f_t$ , en cualquiera de los dos casos, entonces se realiza una transformación de “encogimiento” (*shrink transformation*) [214].

La transformación de encogimiento calcula los nuevos  $n$  vértices del símplex como:

$$x_j = x_l + \delta(x_j - x_l), \quad j = 0, \dots, n; \quad j \neq l \quad (4.17)$$

En la Figura 4.3 se muestra un diagrama de flujo del algoritmo. El procedimiento de optimización concluye teniendo en cuenta tres condiciones diferentes:

1. Se ha alcanzado la tolerancia permitida para los datos de entrada, es decir, el símplex  $S$  resulta muy pequeño y por tanto las distancias entre sus vértices se encuentran por debajo del nivel de tolerancia.
2. Se ha alcanzado el nivel de tolerancia preconcebido para la función  $f$  en cualquiera de los vértices del actual símplex.

3. No se ha logrado una convergencia y por tanto se alcanza un máximo número de iteraciones predeterminadas.

Como se ha podido apreciar, las transformaciones geométricas del símplex son controladas por los cuatro coeficientes definidos,  $\alpha$  (reflexión),  $\beta$  (contracción),  $\gamma$  (expansión) y  $\delta$  (encogimiento), los cuales deben satisfacer las siguientes restricciones [214]:

$$\alpha > 0, 0 < \beta < 1, \gamma > 1, \gamma > \alpha, 0 < \delta < 1 \quad (4.18)$$

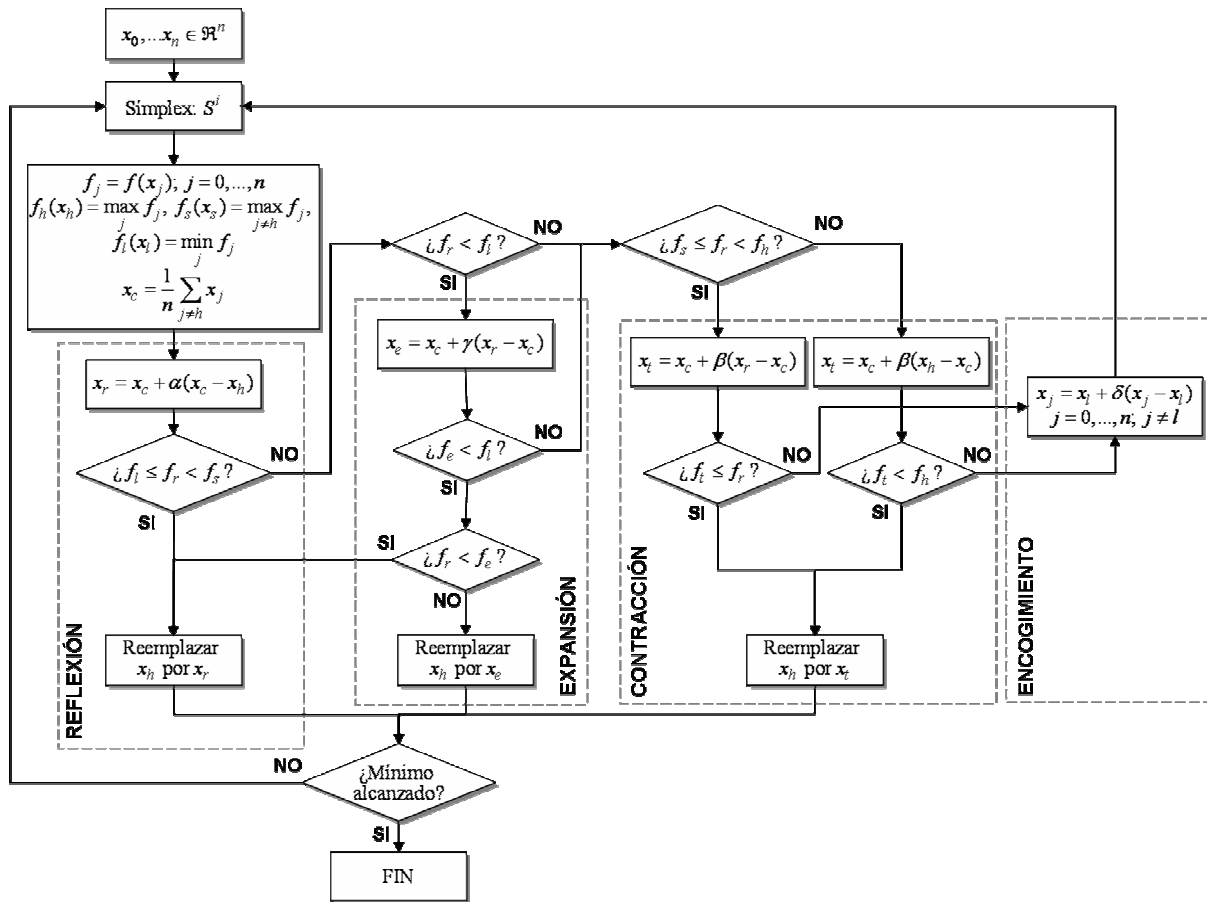


Figura 4.3. Diagrama de flujo del algoritmo de Nelder-Mead.

En la práctica, estos coeficientes suelen ajustarse a unos valores estándares que, a la larga, son los más utilizados en las aplicaciones del algoritmo:

$$\alpha = 1, \beta = \frac{1}{2}, \gamma = 2, \delta = \frac{1}{2} \quad (4.19)$$

Entre las ventajas principales que ofrece el algoritmo de Nelder-Mead, además de ser un método fácil de entender y aplicar, permite obtener un resultado del problema a optimizar con



un número relativamente pequeño de evaluaciones de la función objetivo en cada iteración. Por otro lado, su principal desventaja está relacionada con la convergencia y la influencia de las condiciones iniciales, ya que el método puede ejecutar una gran cantidad de repeticiones sin una mejora significativa en la función objetivo, a pesar de encontrarse lejos de un valor mínimo.

### 4.3 Estudio del algoritmo de optimización temple simulado para su empleo en el ajuste de sistemas de control de procesos complejos

Temple simulado (SA, *Simulated Annealing*) es un método de búsqueda meta-heurística para la resolución de problemas de optimización global. El método simula el proceso físico de templado de materiales, que consiste en el calentamiento de estos materiales y luego su lento enfriamiento para aumentar el tamaño de sus cristales, reducir sus defectos y, de esa forma, minimizar la energía de todo el sistema.

El método fue propuesto de forma independiente por S. Kirkpatrick, C. D. Gelatt y M. P. Vecchi, del “*IBM Thomas J. Watson Research Center*”, en 1983 [215] y por Vlado Černý, del Instituto de Física y Biofísica de la Universidad de Comenius (Bratislava), en 1985 [17]. El algoritmo es una adaptación del algoritmo de Metropolis-Hastings [222, 223], que es un método de tipo Monte Carlo para generar muestras de estados de un sistema termodinámico.

Básicamente, el algoritmo en cada iteración genera un punto de forma aleatoria y calcula la distancia entre el nuevo punto y el anterior mediante una distribución de probabilidad con una escala proporcional a la temperatura. El algoritmo acepta todos los puntos nuevos que reducen el objetivo deseado y también los puntos que lo incrementan con una cierta probabilidad. Al aceptar los puntos que incrementan el objetivo, el algoritmo evita quedarse atrapado en mínimos locales permitiendo, así, una mayor exploración del espacio de búsqueda por más posibles soluciones. El algoritmo, según su avance, selecciona una planificación del templado para disminuir la temperatura de forma sistemática. Con el decremento de la temperatura el algoritmo reduce el espacio de búsqueda hasta converger a un mínimo [224].

El método de temple simulado consiste principalmente de tres relaciones:

1.  $g_T(x)$ : Densidad de probabilidad del espacio de estado de  $D$  parámetros  $x = \{x^i; i = 1, \dots, D\}$ , refiriéndose a  $T$  como la temperatura.
2.  $h(\Delta E)$ : A partir de la variación de la energía obtenida como la diferencia entre el valor actual y el anterior, se calcula la probabilidad de aceptación de una nueva función de coste.
3.  $T(k)$ : Planificación de la “temperatura de templado”  $T$  en  $k$  pasos de tiempo de templado.

La probabilidad de aceptación está basada en la oportunidad de un nuevo estado con “energía”  $E_{k+1}$  relativo al estado previo con “energía”  $E_k$ :

$$h(\Delta E) = \frac{e^{-E_{k+1}/T}}{e^{-E_{k+1}/T} + e^{-E_k/T}} = \frac{1}{1 + e^{\Delta E/T}} \approx e^{-\Delta E/T}; \Delta E = E_{k+1} - E_k \quad (4.20)$$

La ecuación anterior no es más que la distribución de Boltzmann, siendo  $\Delta E$  la diferencia de energía entre estados consecutivos. El valor de energía en cada instante se determina mediante la función de coste a minimizar.

Considerando un conjunto de estados  $x$  con una energía  $e(x)$ , un conjunto de distribuciones de probabilidad  $p(x)$  y la distribución de energía por estado  $d(e(x))$ , el algoritmo también puede ser descrito a partir de la energía total  $E$ :

$$E = \sum_x p(x) d(e(x)) \quad (4.21)$$

A partir de un estado de referencia  $\bar{x}$ , el valor máximo de entropía  $S$  del sistema puede obtenerse como

$$S = - \sum_x p(x) \ln(p(x) / p(\bar{x})) \quad (4.22)$$

El empleo de multiplicadores de Lagrange para limitar la energía al valor medio  $T$ , conduce a la distribución más probable de Gibbs  $G(x)$  normalizada

$$G(x) = \frac{1}{Z} e^{-H(x)/T}; Z = \sum_x e^{-H(x)/T} \quad (4.23)$$

donde  $Z$  es la función de partición y  $H$  es el operador Hamiltoniano utilizado como función de energía.

Para las distribuciones de los estados (4.23) introducidas anteriormente y probabilidades de aceptación del tipo como la definida en (4.20), se mantiene el principio de equilibrio del balance detallado. Por ejemplo, dada las distribuciones de estados consecutivos  $G(x_k)$  y  $G(x_{k+1})$ , si se aplica el criterio de aceptación  $h(\Delta E) = h(E_{k+1} - E_k)$ , estas son iguales:

$$G(x_k) h(\Delta E(x)) = G(x_{k+1}) \quad (4.24)$$

Lo anterior es suficiente para establecer, al menos en teoría, que todos los estados del sistema pueden ser muestreados. Sin embargo, la planificación del templado interrumpe el

equilibrio cada vez que la temperatura cambia y de esta forma, en el mejor de los casos, esto debe realizarse cuidadosamente y de forma gradual.

Un aspecto importante en el algoritmo de temple simulado es la selección del rango de los parámetros a ser buscados. En la práctica, el cálculo de sistemas continuos requiere de su discretización por lo que puede asumirse que, sin mucha pérdida de información, el espacio de trabajo puede ser discretizado. Existen ciertas restricciones adicionales que son necesarias cuando se utilizan funciones de coste con valores integrales. En varias aplicaciones se suelen utilizar técnicas para reducir el espacio inicial a medida que progresa la búsqueda. Por ejemplo, para muchos sistemas físicos, pertenecientes a la clase de procesos Gauss-Markov, es posible escoger como función de densidad de probabilidad del espacio de estado de los parámetros la siguiente distribución [224] de Boltzmann:

$$g(\Delta x) = (2\pi T)^{-D/2} e^{-\Delta x^2/(2T)}; \Delta x = x - x_0 \quad (4.25)$$

tomándose  $\Delta x$  en la práctica como la desviación entre dos estados consecutivos y siendo, además,  $T$  una medida de las fluctuaciones de la distribución  $g$  en el espacio de dimensión  $D$  de  $x$ . A partir de  $g(\Delta x)$ , se ha demostrado que basta con obtener un mínimo global de la energía  $E(x)$  si se selecciona  $T$  que varíe de la forma

$$T(k) = T_0 / \ln k \quad (4.26)$$

siendo  $T_0$  la temperatura inicial y seleccionada lo suficientemente grande. Usualmente, también suele realizarse la planificación de la temperatura de forma exponencial [224], es decir:

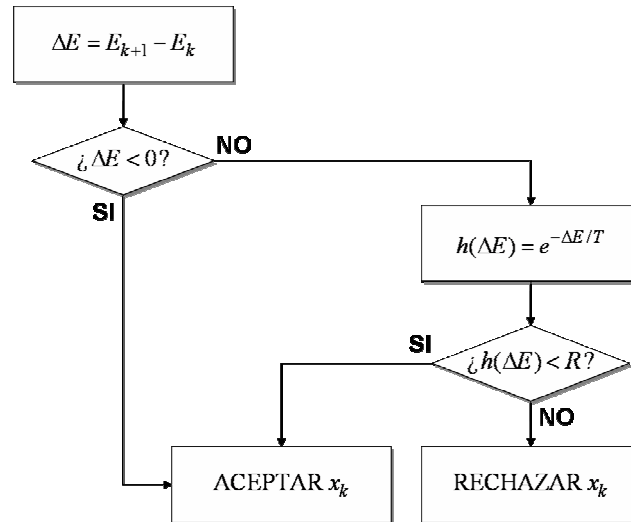
$$\begin{aligned} T(k) &= T_0 e^{(\alpha-1)k} \\ T(k+1) &\approx \alpha T(k) \end{aligned} \quad ; \quad 0 < \alpha < 1 \quad (4.27)$$

donde  $\alpha$  es una constante de reducción cuyos valores típicos se ajustan entre 0.8 y 0.99. Por otra parte, para una variación rápida de la temperatura  $T$  [224], también se utiliza con frecuencia

$$T(k) = T_0 / k \quad (4.28)$$

A partir del cálculo de la variación de energía  $\Delta E$  y de la probabilidad de aceptación  $h(\Delta E)$ , definida en (4.20), en aplicaciones prácticas del algoritmo se utilizan criterios de aceptación-rechazo del estado actual  $x_k$  empleando el algoritmo de Metropolis [222] y que puede resumirse según el diagrama de flujo de la Figura 4.4. Según este criterio, la probabilidad de aceptación se compara con un número  $R$  entre 0 y 1 generado de forma

aleatoria, mediante una distribución uniforme, evitando de esta manera que ocurra una convergencia prematura hacia un mínimo local y se pueda continuar la búsqueda del mínimo global.



**Figura 4.4.** Algoritmo de Metropolis de aceptación-rechazo.

El algoritmo de temple simulado se ejecuta de forma iterativa generando nuevos estados  $x$  que minimicen la energía  $E$  del sistema (función objetivo). Su ejecución terminará teniendo en cuenta varias condiciones diferentes:

1. El nivel de energía  $E$  es menor o igual que cierto valor prefijado límite.
2. Se ha obtenido un nivel de tolerancia preconcebido para la variación de energía  $\Delta E$ .
3. Se ha alcanzado un máximo número de iteraciones predeterminadas.
4. Se ha sobrepasado el tiempo máximo permitido de ejecución del algoritmo.

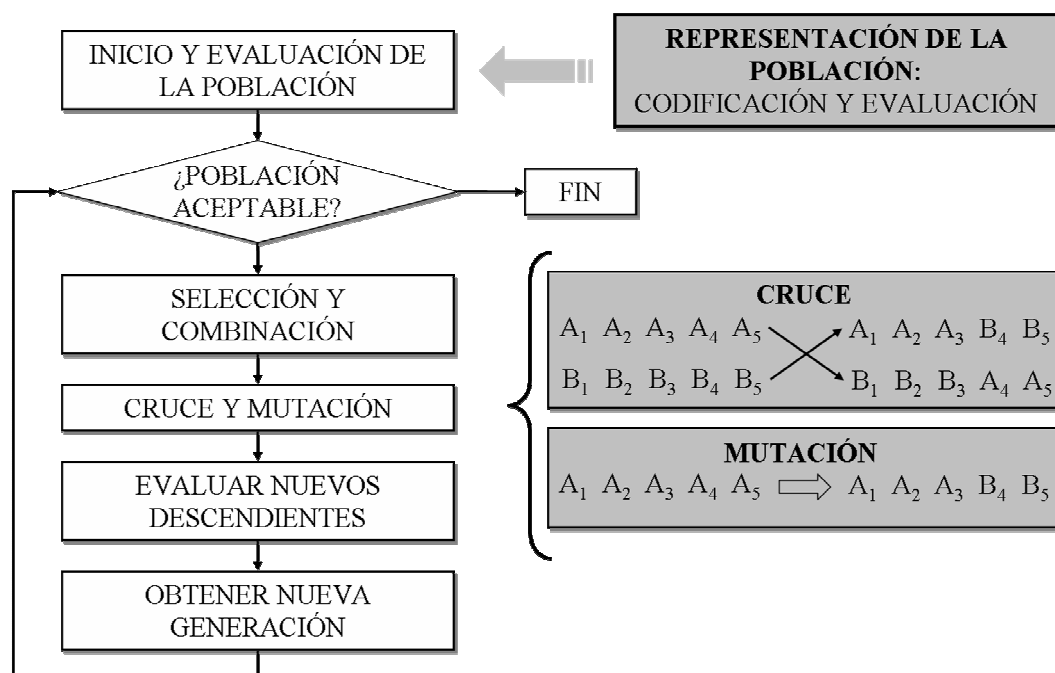
#### 4.4 Breve introducción al estudio de algoritmos genéticos

Los algoritmos genéticos (GA) constituyen un método de búsqueda heurística que imita el proceso de la evolución natural. El método es empleado con frecuencia para generar soluciones a los problemas de búsqueda y optimización [36]. Los algoritmos genéticos pertenecen al grupo de los algoritmos evolutivos (EA), que a su vez son un sub-conjunto del campo computación evolutiva de la inteligencia artificial. Los algoritmos evolutivos realizan una optimización metaheurística basada en población, mediante técnicas inspiradas en la evolución natural, como la herencia, mutación, selección y cruce, para la búsqueda de soluciones a problemas complejos.

El principio básico de funcionamiento del método consiste en la modificación iterativa de una población de soluciones individuales. En cada paso el algoritmo selecciona de forma aleatoria individuos de la población actual para ser padres y ser utilizados para producir una nueva generación. A través de los descendientes de generaciones sucesivas la población "evoluciona" hacia una solución óptima. El algoritmo puede aplicarse para resolver una variedad de problemas de optimización que no son muy adecuados para algoritmos de optimización estándares. Por ejemplo, problemas en los que la función objetivo es discontinua, no diferenciable, estocástica o altamente no lineal.

El algoritmo en cada etapa de creación de una nueva generación a partir de la población actual, utiliza principalmente tres reglas:

- Reglas de selección: seleccionar los individuos, denominados padres, que contribuyen a la población en la próxima generación.
- Reglas de cruzamiento: combinar dos padres para crear los individuos de la generación siguiente.
- Reglas de mutación: aplicar cambios aleatorios a los padres individuales para crear los nuevos individuos.



**Figura 4.5. Diagrama de flujo de datos del algoritmo.**  
(Sanz Bobi, 2001 [225])

El algoritmo fue inicialmente propuesto por el profesor John H. Holland de la Universidad Ann Harbor de Michigan, en 1975 [216]. El Prof. Holland es considerado uno de los pioneros

en el estudio de los sistemas complejos y de la ciencia no lineal. En la actualidad las aplicaciones son muy variadas, abarcando áreas como la bioinformática, la biología, ciencia de la computación, la ingeniería, las ciencias económicas, entre otras. En la Figura 4.5 se muestra un diagrama del algoritmo genético [225] propuesto inicialmente por Holland.

Los miembros de una población se representan por una serie de cromosomas y en la propuesta original del algoritmo son cadenas binarias. La **representación** de cada elemento de la población juega un papel clave en la búsqueda de la solución, ya que ha de ser capaz de representar toda la variedad posible de soluciones del problema en cuestión, así como las características que debe tener la solución final. Una vez determinada la representación, se genera una población inicial y mediante la operación de **selección** se escogen subconjuntos de individuos (generalmente parejas) para ser combinados y producir nuevos individuos (descendencia) [226].

Para crear la nueva generación, el algoritmo realiza una operación de cruce o **cruzamiento** de tal forma que en la nueva población se mantengan ciertas características de los cromosomas de la población padre. La utilidad de esta operación se basa en el hecho de que diferentes componentes de la solución óptima pueden ser evolucionadas independientemente y luego ser combinadas para formar mejores soluciones. Adicionalmente, se utiliza un operador de **mutación** que se ocupa de mantener un nivel de diversidad adecuado en cada población. A continuación se realiza un resumen de los elementos que conforman las etapas del algoritmo genético simple.

**Tabla 4.1. Algunos tipos de representación de los individuos de una población.**

Tipo de representación	Ejemplo						
<i>Binaria</i> : cada gen toma valores binarios, 0 o 1.	<table><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr></table>	1	0	1	1	0	1
1	0	1	1	0	1		
<i>Entera</i> : los genes pueden ser representados por un valor entero.	<table><tr><td>1</td><td>0</td><td>3</td><td>-1</td><td>0</td><td>4</td></tr></table>	1	0	3	-1	0	4
1	0	3	-1	0	4		
<i>Real</i> : cada gen es un valor real.	<table><tr><td>1.78</td><td>2.6</td><td>7</td><td>0</td><td>-1.2</td><td>6.5</td></tr></table>	1.78	2.6	7	0	-1.2	6.5
1.78	2.6	7	0	-1.2	6.5		

**Formas de representación:** Como se ha comentado con anterioridad, inicialmente se utilizaban cadenas binarias, debido a que se adaptan adecuadamente a las operaciones que se utilizan en el algoritmo, pero no resultan ser eficientes del todo por lo que con frecuencia

suelen utilizarse otro tipos de representaciones, como son las representaciones mediante valores enteros y valores reales (ver Tabla 4.1).

**Tabla 4.2. Ejemplos de operadores de cruce.**

Función de cruce	Ejemplo																											
<i>Aleatorio</i> : se genera un vector binario de genes a seleccionar de cada padre.	<p>Sea el siguiente vector binario generado de forma aleatoria, donde 1 se corresponde con los genes del padre <math>P_1</math> y 0 los genes del padre <math>P_2</math>:</p> <table><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr></table> <p>Sean además, los vectores de genes de los padres:</p> <table><tr><td><math>P_1</math>:</td><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td><td>f</td></tr><tr><td><math>P_2</math>:</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr></table> <p>Al combinar los genes de los padres, se obtiene el siguiente vector de genes del individuo hijo:</p> <table><tr><td><math>H_1</math>:</td><td>a</td><td>1</td><td>c</td><td>d</td><td>5</td><td>f</td></tr></table>	1	0	1	1	0	1	$P_1$ :	a	b	c	d	e	f	$P_2$ :	1	2	3	4	5	6	$H_1$ :	a	1	c	d	5	f
1	0	1	1	0	1																							
$P_1$ :	a	b	c	d	e	f																						
$P_2$ :	1	2	3	4	5	6																						
$H_1$ :	a	1	c	d	5	f																						
<i>Un punto</i> : se genera un número aleatorio que representa una posición en la cadena de genes de los padres y se intercambian los genes a la izquierda de esta posición.	<p>Sean <math>P_1</math> y <math>P_2</math> cadenas de genes de los progenitores:</p> <table><tr><td><math>P_1</math>:</td><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td><td>f</td></tr><tr><td><math>P_2</math>:</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr></table> <p>y <math>n = 4</math> el punto de cruzamiento, generado de forma aleatoria, al combinar los genes de los padres se obtiene la siguiente cadena de genes del individuo hijo:</p> <table><tr><td><math>H_1</math>:</td><td>a</td><td>b</td><td>c</td><td>d</td><td>5</td><td>6</td></tr></table>	$P_1$ :	a	b	c	d	e	f	$P_2$ :	1	2	3	4	5	6	$H_1$ :	a	b	c	d	5	6						
$P_1$ :	a	b	c	d	e	f																						
$P_2$ :	1	2	3	4	5	6																						
$H_1$ :	a	b	c	d	5	6																						
<i>Cruce entre 2 puntos</i> : se generan dos números aleatorios que representan dos posiciones en la cadena de genes de los padres y se intercambian los genes entre estas dos posiciones.	<p>Sean <math>P_1</math> y <math>P_2</math> cadenas de genes de los progenitores:</p> <table><tr><td><math>P_1</math>:</td><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td><td>f</td></tr><tr><td><math>P_2</math>:</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr></table> <p>y <math>m = 2</math>, <math>n = 4</math> los puntos de cruzamiento generados de forma aleatoria, al combinar los genes de los padres se obtiene la siguiente cadena de genes del individuo hijo:</p> <table><tr><td><math>H_1</math>:</td><td>a</td><td>b</td><td>3</td><td>4</td><td>e</td><td>f</td></tr></table>	$P_1$ :	a	b	c	d	e	f	$P_2$ :	1	2	3	4	5	6	$H_1$ :	a	b	3	4	e	f						
$P_1$ :	a	b	c	d	e	f																						
$P_2$ :	1	2	3	4	5	6																						
$H_1$ :	a	b	3	4	e	f																						

**Población inicial:** Esta puede generarse de varias formas, una de la más utilizada es generar los individuos o sus genes, empleando una distribución uniforme. Goldberg ofrece una colección más extensa de funciones o formas para generar la población inicial [36].

**Evaluación de la población:** La función de evaluación ha de medir la adaptación de cada uno de los individuos dependiendo, fundamentalmente, del tipo de representación elegido ya que esta función evalúa la información contenida en los genes. Si se emplea el algoritmo para resolver problemas de optimización, la función a optimizar se convierte en la función de evaluación.

**Operación de selección:** Mediante esta operación se transmiten aquellas características de las soluciones que son consideradas valiosas a lo largo de generaciones. Los individuos mejor evaluados tienen más probabilidades de reproducirse. Algunas de las funciones más utilizadas para esta etapa son la selección por ruleta [36], por torneo [227] y la estocástica uniforme, entre otras.

**Cruzamiento:** En esta etapa, la información de la población se combina para crear mejores individuos. Algunos de los operadores de mayor utilización (ver Tabla 4.2) resultan el cruzamiento aleatorio, el cruce de un punto y entre 2 puntos, entre otros.

**Mutación:** Mediante la mutación el algoritmo realiza pequeños cambios aleatorios en los genes de los individuos de la población actual y así fomentar la variabilidad dentro de la población. Entre los operadores más utilizados está la función Gaussiana que adiciona números aleatorios, tomados de una distribución normal con media igual a 0, a cada gen del individuo. Otro operador es la función de mutación uniforme donde, primeramente, se selecciona una parte de los genes de un individuo asignándole a cada gen una razón de probabilidad de ser mutado y, seguidamente, el algoritmo reemplaza cada gen seleccionado por un número aleatorio que esté comprendido dentro del rango del gen dado.

**Reemplazo de la población:** En esta etapa, los nuevos individuos creados durante la etapa de cruzamiento reemplazarán a parte de la población existente para crear la nueva generación. El reemplazo puede realizarse mediante diferentes métodos. Algunos de los más comúnmente empleados son el reemplazo de un número de individuos que se corresponde con los peores evaluados de la población, cada nuevo individuo reemplaza al peor evaluado y cada nuevo individuo sustituye a cualquier individuo elegido de forma aleatoria.

**Condición de parada del algoritmo:** Existen varias condiciones de paradas del algoritmo, que pueden ser, por ejemplo, alcanzar un número máximo de generaciones, un tiempo máximo de ejecución del algoritmo o, en el caso de optimización o ajuste de funciones, cuando se alcance un error igual o menor a un error mínimo límite, entre otras condiciones de parada.



Todas las etapas anteriormente descritas del algoritmo hacen que su robustez sea superior comparada con otras técnicas clásicas de búsqueda y optimización. Se resaltan como características principales al empleo de una población de puntos del espacio de búsqueda y no de un único punto y el uso de mecanismos de elección aleatoria como herramienta de búsqueda de mejores regiones del espacio. Además, se debe destacar su característica de blindaje, ya que la única información que toman del entorno es el valor de la función objetivo del conjunto de parámetros, es decir, no emplean información específica del problema [225].

Información más exhaustiva acerca de las diferentes etapas del algoritmo, así como los diferentes operadores que se utilizan en cada una de ellas además de propuestas de variaciones del algoritmo o hibridaciones con otras técnicas de optimización, pueden encontrarse en los trabajos de Holland [216, 228], Goldberg [36, 227], Grefenstette [217, 229], entre otros.

#### 4.5 Estudio del método de optimización de entropía cruzada

El método de entropía cruzada (CE) [20] representa un enfoque general de tipo Monte Carlo para la optimización combinatorial, continua y con multiplicidad de extremos. El método fue introducido, inicialmente, para simular eventos raros con probabilidades muy pequeñas pero con requerimientos de una buena precisión en su estimación, como es el caso de las redes estocásticas. Posteriormente, fue adaptado para la optimización de sistemas asumiendo que el muestreo en todo el óptimo de una función tiene una probabilidad muy pequeña.

Sea  $x$  una variable aleatoria, en el espacio  $\mathcal{X}$ , con una función de densidad de probabilidad (FDP)  $p_x$  y  $\phi(x)$  una función real definida en el espacio  $\mathcal{X}$ . El objetivo del método de entropía cruzada es encontrar el mínimo de la función  $\phi(x)$  y los correspondientes estados  $x^*$  que satisfacen este mínimo:

$$\gamma^* = \phi(x^*) = \min_{x \in \mathcal{X}} \phi(x) \quad (4.29)$$

El método CE provee una metodología para crear una secuencia  $x_0, x_1, \dots, x_N$  y de niveles  $\gamma_0, \gamma_1, \dots, \gamma_N$ , tal que  $\underline{\gamma}$  converge a  $\gamma^*$  y  $x$  converge a  $x^*$ .

El principal problema consiste en estimar la probabilidad  $l(\gamma)$  de un evento  $E_\gamma = \{x \in \mathcal{X} | \phi(x) \geq \gamma\}, \gamma \in \mathfrak{R}^+$ .

Definiendo una colección de funciones para  $x \in \mathcal{X}, \gamma \in \mathfrak{R}^+$

$$I_v(x, \gamma) = I_{\{\phi(x_i) > \gamma\}} = \begin{cases} 1 & \text{if } \phi(x) \leq \gamma \\ 0 & \text{if } \phi(x) > \gamma \end{cases} \quad (4.30)$$

Sea  $g(-, \nu)$  una familia de funciones de densidad de probabilidad en el espacio  $\mathcal{X}$  parametrizadas por un vector de valores reales  $\underline{\nu} \in \Re^+$  y  $g(x, \nu) | \nu \in \Lambda$ .

$$\begin{aligned} l(\gamma) &= P_v(\phi(x) \geq \gamma) \\ &= E_v \cdot I_v(x, \nu) \end{aligned} \quad (4.31)$$

donde  $E_v$  representa el correspondiente operador de esperanza.

De esta forma la ecuación (4.31) convierte el problema de optimización en un problema estocástico asociado con una probabilidad muy pequeña. Se utiliza una técnica de minimización de varianza, tal como el muestreo de importancia, donde la muestra aleatoria es elaborada, *a priori*, a partir de la función de densidad de probabilidad  $h$  apropiada. Tomando una muestra aleatoria  $x_0, x_1, \dots, x_N$ , a partir de un muestreo de importancia (diferente) de densidad  $h$  en  $\mathcal{X}$  y evaluando:

$$\hat{l} = 1/N \cdot \sum_{i=1}^N I_{\{\phi(x_i) > \gamma\}} \cdot W(x_i) \quad (4.32)$$

donde  $\hat{l}$  es el muestreo de importancia y  $W(x) = g(x, \nu) / h(x)$  se denomina la razón de verosimilitud.

La búsqueda de la densidad de muestreo  $h^*(x)$  resulta de gran dificultad, ya que la determinación de  $h^*(x)$  requiere que  $l$  sea conocido.

$$h^*(x) = \frac{I_{\{\phi(x_i) \geq \gamma\}} \cdot g(x, \nu)}{l} \quad (4.33)$$

Así el vector de parámetros, denominado parámetro referenciado  $\nu^*$ , debe ser escogido de tal forma que la distancia entre  $h^*$  y  $g(x, \nu)$  es mínima, por tanto se reduce el problema a un caso escalar.

Una medida de la distancia entre dos densidades  $g$  y  $h$  es la distancia de Kullback-Leibler, también denominada entropía cruzada entre  $g$  y  $h$ :

$$D(g, h) = \int g(x) \cdot \ln g(x) dx - \int g(x) \cdot \ln h(x) dx \quad (4.34)$$

La minimización de  $D(g(x, \nu), h^*)$  es equivalente a maximizar  $\int h^*(x) \cdot \ln g(x, \nu) dx$ , lo cual implica:

$$\max_{\nu} D(\nu) = \max_{\nu} E_p \left( I_{\{\phi(x_i) \geq \gamma\}} \cdot \ln g(x, \nu) \right) \quad (4.35)$$

Empleando nuevamente el muestreo de importancia, se puede reescribir la ecuación (4.35) para calcular la esperanza en (4.35). Por tanto se puede elaborar una muestra  $x_1, x_2, \dots, x_N$  a partir de  $g$  y estimar el máximo (o el mínimo) de  $\hat{D}(\nu)$ :

$$\max_{\nu} \hat{D}(\nu) = \max_{\nu} \frac{1}{N} \sum_{i=1}^N I_{\{\phi(x_i) \geq \gamma\}} \cdot \frac{p_x(x_i)}{h(x_i)} \cdot \ln g(x_i, \nu) \quad (4.36)$$

Sin embargo  $h$  continúa siendo desconocido en la ecuación (4.36). El algoritmo de CE trata de superar esta dificultad mediante la construcción adaptativa de una secuencia de parámetros  $(\gamma_t | t \geq 1)$  y  $(\nu_t | t \geq 1)$ .

#### 4.5.1 Algoritmo de entropía cruzada adaptado para la solución de problemas de optimización

La esencia del método CE adaptado para resolver problemas de optimización, se describe a continuación. Si se considera el siguiente problema de optimización:

$$\phi(x^*) = \gamma^* = \max_{x \in \mathcal{X}} \phi(x) \quad (4.37)$$

El principio del método CE para optimización, radica en convertir el problema planteado en la ecuación (4.37) en un problema estocástico asociado y resolverlo de forma adaptativa, como la simulación de un evento raro. Si  $\gamma^*$  es el óptimo de  $\phi$ , la cuestión principal reside en definir una familia  $g(\cdot, \nu) | \nu \in \Lambda$  e iterar el algoritmo CE con el fin de que  $\gamma_t \xrightarrow{\infty} \gamma^*$ , obteniendo muestras alrededor del óptimo.

El algoritmo puede resumirse como sigue:

1. Inicializar  $\nu_t = \nu_0$
2. Generar un conjunto de muestras de tamaño  $N(x_t^i)_{1 \leq i \leq N}$  a partir de  $g(x, \nu_t)$ , calcular  $\phi(x_t^i)$ , y ordenar  $\phi_1 \geq \phi_2 \geq \dots \geq \phi_N$  de mayor a menor. Utilizar  $\gamma_t = \phi_{[\rho N]}$  para seleccionar el sub-conjunto de población élite.

3. Actualizar  $v_t$  mediante:

$$v_{t+1} = \arg \min_v \frac{1}{N^{elite}} \sum_{i=1}^{N^{elite}} I_{\{\phi(x_i^t \geq \gamma_t)\}} \cdot \ln g(x_i^t, v_t) \quad (4.38)$$

4. Repetir desde el paso 2 hasta que se produzca una convergencia o se cumpla algún criterio de finalización del algoritmo.
5. Si se asume que la convergencia se alcanza en  $t = t^*$ , un valor óptimo de  $\phi$  puede obtenerse a partir de  $g(., v_{t^*})$ .

El tercer paso de actualización de parámetros se realiza utilizando las muestras de mejor desempeño  $N^{elite} = \rho \cdot N$ , también denominadas muestras élite. Los parámetros de actualización son encontrados mediante estimaciones de máxima verosimilitud (EMV) de la muestra élite [230]. La distribución de las muestras puede ser bastante arbitraria y no necesita estar relacionada a la función a optimizar [231]. El empleo de la función de distribución normal (Gaussiana) ofrece una formulación de actualización fácil y simple.

Considerando un problema de optimización de  $n$ -dimensional, los vectores de media  $\underline{\mu}$  y varianza  $\underline{\sigma}$  de las  $n$  distribuciones son estimados en cada iteración  $t$  para cada componente  $j = 1, \dots, n$  de los vectores como sigue:

$$\tilde{\mu}_{ij} = \sum_{i=1}^{N^{elite}} x_{ij} / N^{elite} \quad (4.39)$$

$$\tilde{\sigma}_{ij}^2 = \sum_{i=1}^{N^{elite}} (x_{ij} - \mu_{ij})^2 / N^{elite} \quad (4.40)$$

donde  $4 \leq N^{elite} \leq 20$  son valores típicos.

El vector de medias  $\hat{\underline{\mu}}$  debe converger a  $\gamma^*$  y el vector de desviación estándar  $\hat{\underline{\sigma}}$  debe converger al vector de ceros. Con el objetivo de prevenir la ocurrencias de unos y ceros en los vectores de parámetros, se introduce un factor de suavizado  $\alpha$  para el vector de medias y de suavizado dinámico  $\beta_t$  para la varianza

$$\hat{\underline{\mu}}_t = \alpha \cdot \tilde{\underline{\mu}}_t + (1 - \alpha) \cdot \hat{\underline{\mu}}_{t-1} \quad (4.41)$$

$$\hat{\underline{\sigma}}_t = \beta_t \cdot \tilde{\underline{\sigma}}_t + (1 - \beta_t) \cdot \hat{\underline{\sigma}}_{t-1} \quad (4.42)$$

$$\beta_t = \beta - \beta \cdot \left(1 - \frac{1}{t}\right)^q \quad (4.43)$$

donde  $0.4 \leq \alpha \leq 0.9$  son valores típicos,  $\beta$  es una constante de suavizado (valores recomendados entre 0.6 y 0.9) y  $q$  es un entero con valores típicos entre 2 y 7.

## 4.6 Conclusiones

En este capítulo se ha realizado un estudio de métodos de optimización estocásticos y libres de gradiente, de inspiración geométrica, física y biológica, para su aplicación al ajuste óptimo de sistemas de control de procesos complejos. Los métodos de optimización permitirán desarrollar procedimientos de ajuste de parámetros de control. Se trata de las estrategias de control que han sido diseñadas por técnicas de *soft-computing* y clásicas como las descritas en el Capítulo 3 y que utilizarán representaciones computacionales eficientes (modelos) descritos en el Capítulo 2. El fin último es, como veremos más adelante, utilizar estos métodos de optimización en arquitecturas jerárquicas de control inteligente de procesos con una relativa alta complejidad.

En el capítulo siguiente se propone un método de diseño de las estrategias de control para ser utilizadas en el control de procesos de fabricación de alta complejidad, en específico, el proceso de taladrado de alto rendimiento. Entre las etapas de diseño, tiene un papel muy importante el uso de los algoritmos de optimización introducidos en este capítulo, ya que permitirán obtener un alto desempeño de las estrategias de cara al cumplimiento de los requisitos de control.



---

## **CAPÍTULO 5**

# **DISEÑO DE SISTEMAS INTELIGENTES DE CONTROL ÓPTIMOS PARA PROCESOS COMPLEJOS**

---

Este capítulo se centra, principalmente, en el diseño de sistemas de control óptimos, basados en técnicas clásicas y de *soft-computing*, de procesos complejos y su aplicación a un proceso de fabricación, en específico el taladrado de alto rendimiento. La metodología de diseño empleada presenta entre sus principales etapas, la simulación de las estrategias de control desarrolladas a partir de las técnicas propuestas en el Capítulo 3 y su implementación en una plataforma experimental. El diseño de los sistemas de control se realiza a partir de modelos de los procesos físicos complejos obtenidos en el Capítulo 2, en específico, a partir del modelo dinámico, computacionalmente eficiente, del proceso de taladrado de alto rendimiento.

Durante las etapas de desarrollo de los sistemas de control, resulta clave el diseño e implementación de los métodos de optimización con vistas a la sintonía o ajuste óptimos de los sistemas de control y que fueron presentados en el capítulo anterior. Asimismo, gana especial relevancia el análisis de la capacidad de estas estrategias de optimización para ser integradas en sistemas intermediarios con especificaciones de tiempo real, además de su eficiencia computacional. Es por eso que el desarrollo de estudios comparativos entre las estrategias de control y entre los diferentes métodos de optimización, permitirá evaluar el alcance real de algunas de estas técnicas, hecho que resultará esencial para dotar a los sistemas de control inteligente de capacidades *self*-\* (auto-optimización, auto-adaptación, auto-configuración y auto-aprendizaje) más eficientes para el control de procesos complejos y, específicamente, de procesos de fabricación.

El capítulo se divide en siete secciones. La sección 5.1 presenta el diseño e implementación de una estrategia de control lineal para la optimización de un proceso de fabricación complejo, en específico el proceso de taladrado del alto rendimiento. A continuación, en la sección 5.2 se describe el diseño e implementación de un sistema de control por modelo interno, basado en modelos de inferencia neuro-borrosa, para el control de un proceso de fabricación.

Por otra parte, en las cuatro secciones siguientes, se realiza el diseño e implementación de estrategias de optimización, clásicas y contemporáneas, tanto de inspiración geométrica, física como biológica, para el ajuste de un sistema de inferencia borrosa empleado para el control de un proceso de fabricación complejo, como es el caso del taladrado de alto rendimiento. La optimización se realiza mediante diferentes métodos, como son: el algoritmo heurístico de Nelder-Mead (sección 5.3), el método de inspiración física temple simulado (sección 5.4), de inspiración biológica como son los algoritmos genéticos (sección 5.5) y el método de entropía cruzada (sección 5.6). Seguidamente en la sección 5.7 se realiza un estudio comparativo del funcionamiento de las técnicas de optimización empleadas para el ajuste de la estrategia de control borrosa. Por último, se plantean las conclusiones sobre los resultados obtenidos.

### **5.1 Diseño e implementación de una estrategia lineal para la optimización de un proceso de fabricación. Aplicación al caso particular del proceso de taladrado de alto rendimiento**

En esta sección se presentan algunos resultados en el diseño e implementación de un sistema de control jerárquico, del proceso de taladrado de alto rendimiento (TAR) en un centro de mecanizado de alta velocidad [232]. El objetivo del sistema de control es mantener constante la fuerza de corte, como variable controlada, ante el incremento en la profundidad de corte, actuando para ello de forma óptima, sobre la velocidad de avance de la herramienta de corte. Se introduce el procedimiento de creación de todo el sistema de control, el cual está basado en la estrategia de control lineal proporcional, integral y derivativo (PID), cuya función de transferencia se describe en la ecuación (5.1). Para el ajuste y simulación del sistema de control se ha utilizado el modelo computacional del proceso de TAR introducido en la sección 2.3 de este trabajo, que representa un modelo de tercer orden pero con un mayor retardo de tiempo igual a 0.4 s:

$$G_{PID}(s) = \frac{f(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d \cdot s \quad (5.1)$$

$$G(s) = \frac{F(s)}{f(s)} = \frac{10.262e^{-0.4s}}{(0.25s+1)(0.16s+1)(0.13s+1)} \quad (5.2)$$



La estrategia de control PID es ejecutada en un ordenador, dentro de una arquitectura jerárquica como la que se muestra en la Figura 2.6 del Capítulo 2, en la cual las acciones de control son enviadas al controlador numérico (CNC) abierto del centro de mecanizado, a través del bus de campo MPI y la señal de fuerza de corte se obtiene de una plataforma dinamométrica, a través de una tarjeta de adquisición de datos.

Para el diseño del controlador PID se ha utilizado el método clásico de Ziegler-Nichols (Z-N) [233]. Aplicando el método Z-N, a partir de la función de transferencia del proceso (5.2), se calcula primeramente la ganancia crítica ( $K_u = 0.181$ ), la cual se corresponde a la ganancia límite del controlador, sólo con acción proporcional (P), con la cual se producen oscilaciones sostenidas en el sistema de control, siendo  $T_u = 1.611$  s el periodo de las oscilaciones. A partir de estos valores se calculan los parámetros del controlador PID, según Z-N, obteniéndose los siguientes valores: ganancia proporcional  $K_p = 0.1086$  ( $0.6K_u$ ), constante de tiempo de la acción integral  $T_i = 0.8055$  s ( $T_u/2$ ) y constante de tiempo de la acción derivativa  $T_d = 0.201375$  s ( $T_u/8$ ).

### 5.1.1 Simulación del sistema de control

Una vez obtenidos los parámetros de ajuste del controlador PID según Z-N, el sistema de control a lazo cerrado es simulado mediante la aplicación de Matlab/Simulink que se muestra en la Figura 5.1. Adicionalmente, el sistema de control incorpora límites de saturación para la velocidad de avance como acción de control, entre 10 y 130 mm/min ( $10 \leq f \leq 130$  mm/min) y se incorpora, además, un filtro pasa bajo para suavizar los cambios abruptos en la variable de referencia (fuerza de corte de referencia) del sistema de control. Por otra parte, dado que la acción de control es ajustada o varía alrededor de su valor nominal, a la salida del controlador PID se le adiciona un valor constante, correspondiente al valor nominal de la velocidad de avance.

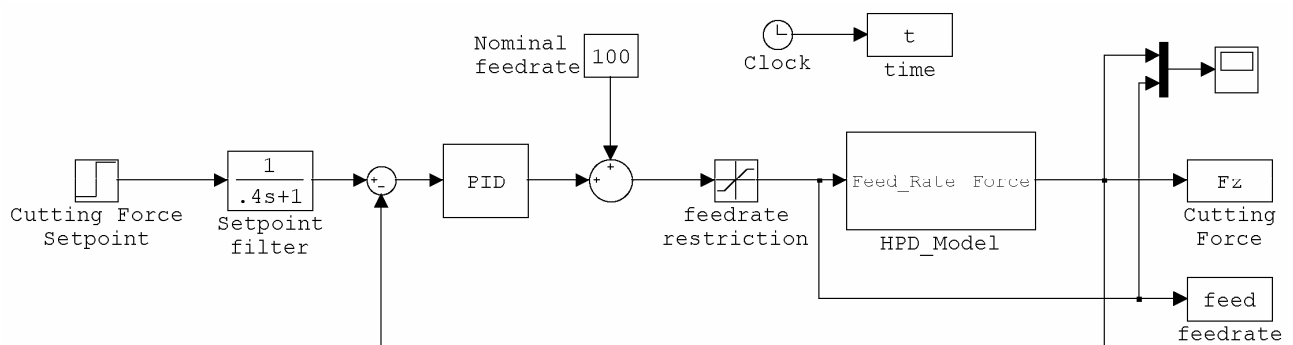


Figura 5.1. Diagrama de simulación del sistema de control de la fuerza de corte.

De la simulación del sistema de control ante cambios de tipo escalón en la referencia, se obtiene un comportamiento en la fuerza de corte con un máximo sobrepaso de un 29.1%. Este comportamiento dinámico del sistema no es deseable para el proceso TAR, por lo que se hace necesario reajustar los parámetros del controlador con el objetivo de minimizar el máximo sobrepaso. Respondiendo a una situación similar, en un trabajo previo, Oh et ál. [163] reducen la ganancia  $K_p$  del controlador hasta que los polos dominantes del sistema a lazo cerrado sean reales e iguales, lo cual se corresponde con una respuesta temporal aperiódica. Empleando este mismo criterio y el método clásico del lugar de las raíces (ver Figura 5.2), se ajusta la nueva ganancia del controlador:  $K_p = 0.000188$ .

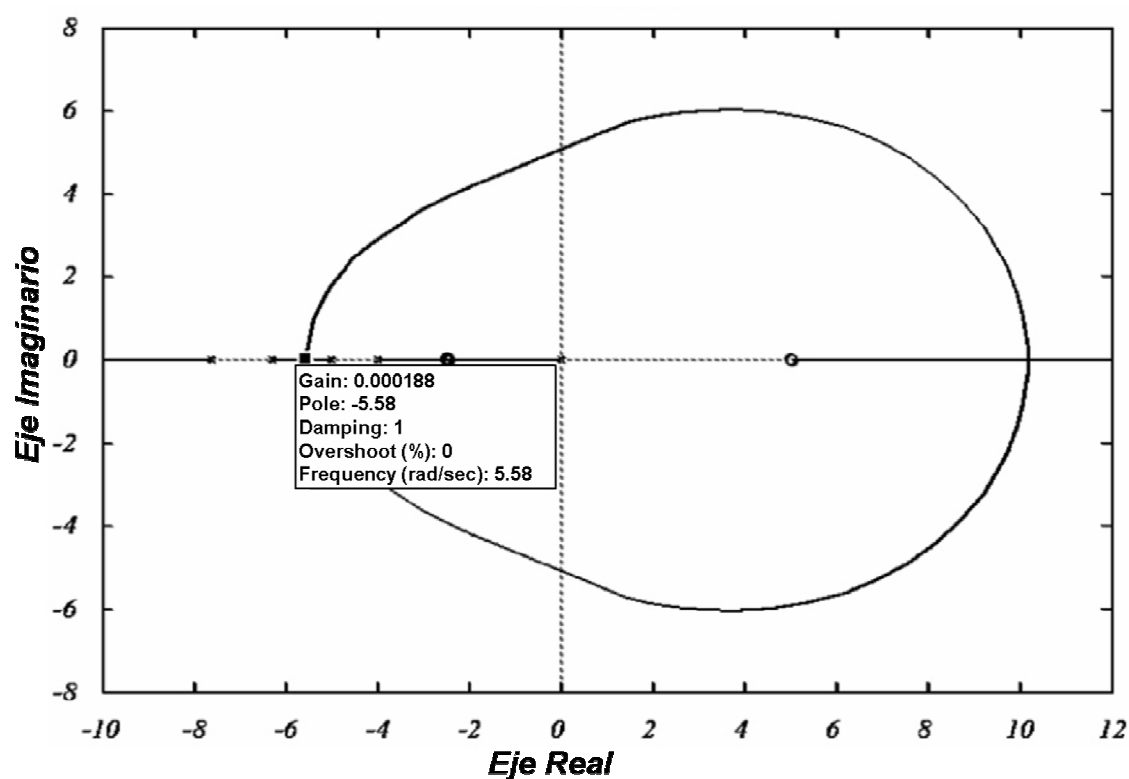


Figura 5.2. Gráfica del lugar de las raíces para una respuesta aperiódica.

Empleando esta nueva ganancia proporcional se ha simulado nuevamente el sistema de control, corroborándose la eliminación del sobrepaso pero, por otra parte, la respuesta transitoria se ralentiza. Nuevamente, de forma similar al procedimiento utilizado por Oh et ál. [161], se ajusta la ganancia mediante el método de ajuste de prueba y error, con el objetivo de minimizar el máximo sobrepaso e incrementar la velocidad de respuesta del sistema. En la Figura 5.3 se muestra la respuesta temporal del sistema como consecuencia de su simulación para diferentes ajustes de la ganancia del controlador PID.

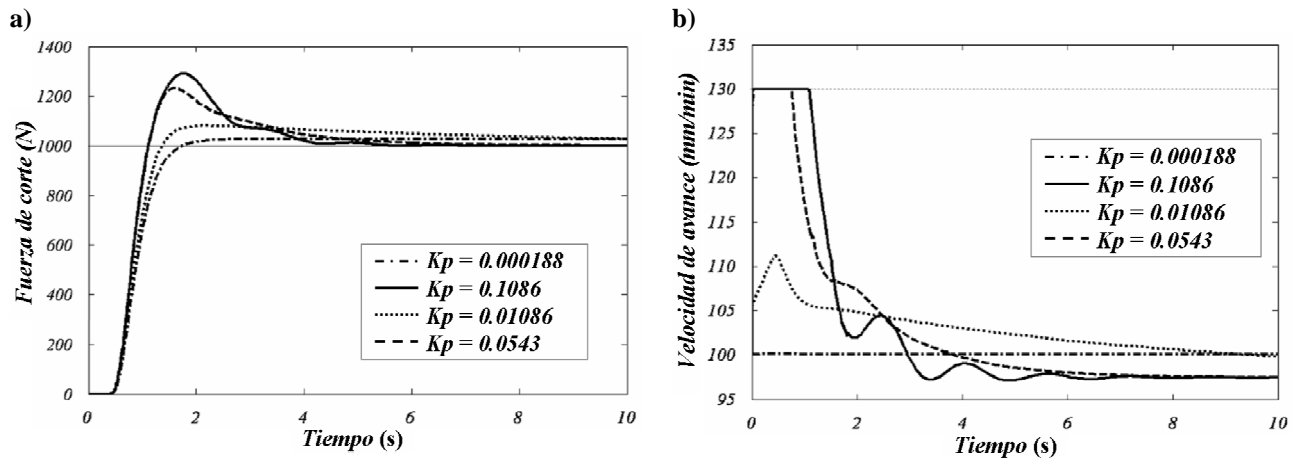


Figura 5.3. Simulación del sistema de control PID a lazo cerrado.  
a) Fuerza de corte y b) velocidad de avance.

### 5.1.2 Implementación experimental del sistema de control

Seguidamente, la siguiente etapa constituye la implementación del controlador en tiempo real en la plataforma experimental. El algoritmo de control se ejecuta en un ordenador, enviándose la acción de control hacia el CNC, a través del bus de campo MPI. El acceso a este bus se realiza mediante una tarjeta de comunicación y la comunicación con ella se realiza vía la interfaz de intercambio dinámico de datos (DDE) de Microsoft. La señal de fuerza de corte es adquirida a una frecuencia de muestreo de 1 kHz y la acción de control es actualizada a una frecuencia de 14.5 Hz.

El controlador se ha implementado mediante un algoritmo PID discreto clásico en Matlab/Simulink, descrita en la ecuación (5.3), y el acceso al sistema de adquisición de datos y al bus MPI, se realiza mediante la herramienta de Matlab, Real-Time Workshop (RTW). El algoritmo calcula la acción de control a partir de la señal de error  $e(t)$  entre la fuerza de referencia (SP) y la fuerza corte medida (Fz). A la señal de salida del controlador se le adiciona el valor nominal de la velocidad de avance y esta señal se limita entre 10 y 130 mm/min para evitar saturaciones en el par de corte de la máquina herramienta (ver Figura 5.4). Por otra parte, se ha insertado un filtro discreto de primer orden pasa bajo (5.4) después de la señal de referencia, evitando de esta forma que se generen cambios abruptos en la acción de control. Además, en el controlador se ha adicionado un valor de umbral en la fuerza de corte para su activación. De esta manera se garantiza que el control sólo se activa si la máquina herramienta está realizando la operación de corte dentro de los valores nominales esperados, es decir, la fuerza de corte es mayor que un valor mínimo esperado.

$$\begin{aligned}
f_p[k] &= K_p e[k] \\
f_i[k] &= f_i[k-1] + (K_p T_s / T_i) e[k] \\
f_d[k] &= (K_p T_d / T_s) (e[k] - e[k-1]) \\
f[k] &= f_p[k] + f_i[k] + f_d[k]
\end{aligned} \tag{5.3}$$

$$F(z) = \frac{1-\tau}{z-\tau} \tag{5.4}$$

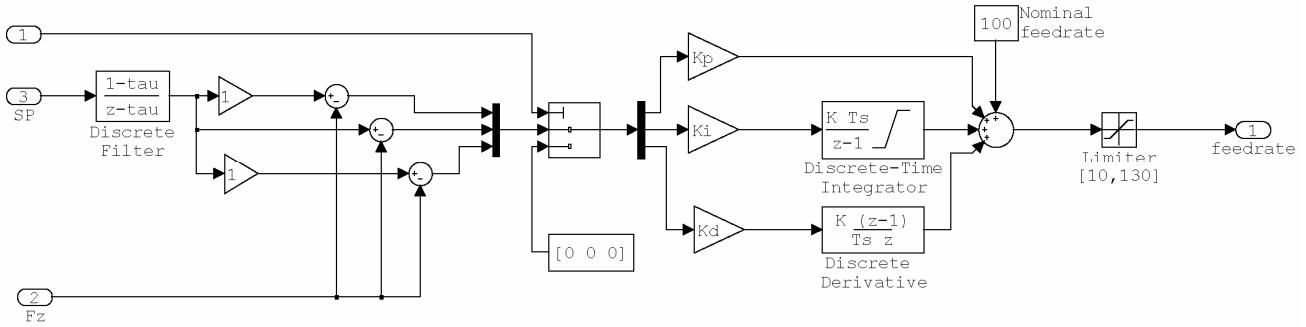


Figura 5.4. Diagrama del controlador PID implementado.

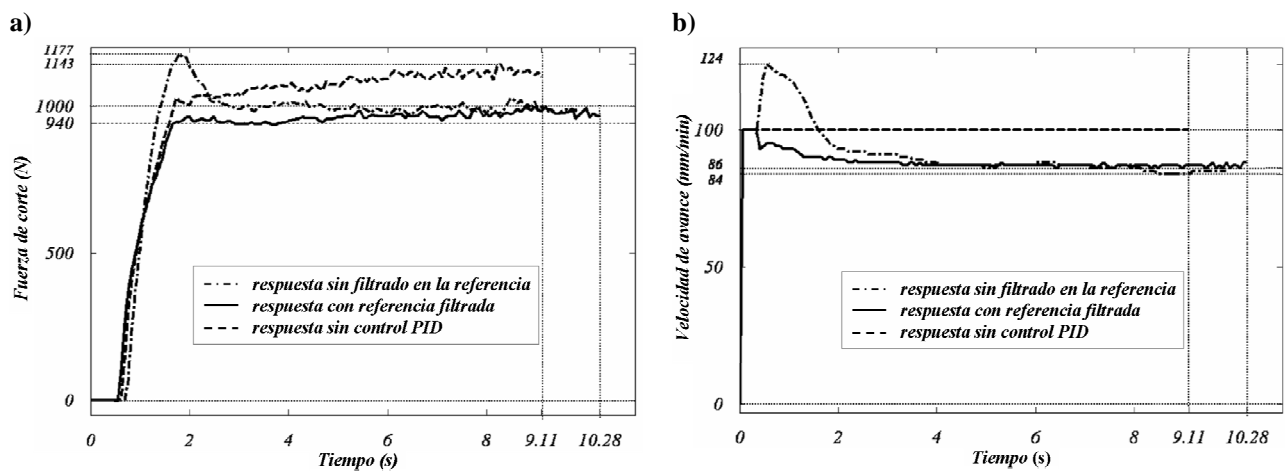
Con vista a probar el desempeño del controlador PID, se han realizado experimentos de control del proceso de TAR a lazo cerrado, empleándose los siguientes parámetros:

- Referencia en la fuerza:  $SP = 1000 \text{ N}$
- Tiempo de muestreo del lazo:  $T_s = 0.069 \text{ s}$
- Coeficiente del filtro:  $\tau = 0.95$
- Velocidad de avance nominal:  $100 \text{ mm/min}$
- Velocidad de giro nominal:  $870 \text{ rpm}$
- Umbral de activación del controlador:  $100 \text{ N}$
- Ganancia de la acción proporcional:  $K_p = 0.01086$
- Ganancia de la acción integral:  $K_i = K_p / T_i = 0.01348$
- Ganancia de la acción derivativa:  $K_d = K_p T_d = 0.00219$
- Límites de saturación de la velocidad de avance:  $[10, 130] \text{ mm/min}$

Como puede observarse, el controlador PID utiliza una versión modificada de los parámetros ajustados inicialmente mediante Z-N. Con el objetivo de mejorar la respuesta transitoria del sistema de control, la ganancia  $K_p$  se ha ajustado a un 10% de su valor calculado inicialmente mediante Z-N. La Figura 5.5 muestra la respuesta en tiempo real del sistema de control, tanto la fuerza de corte controlada, como la velocidad de avance como acción de control. En la figura se muestran tres condiciones de operación diferentes: sin el

control PID activado, es decir, a lazo abierto, control a lazo cerrado sin filtrado en la referencia de la fuerza y también a lazo cerrado pero con el filtro pasa bajo en la referencia.

Como puede observarse en la Figura 5.5, el filtrado del valor de referencia, mediante un filtro discreto de primer orden, mejora el comportamiento dinámico de la fuerza de corte, obteniéndose una respuesta del sistema de control sin sobrepaso, contrario a la respuesta dinámica sin filtrado, en la que se obtiene un máximo sobrepaso de un 17.7 %. El sistema de control es capaz de mantener la fuerza con muy poca variación alrededor de su valor de referencia, durante toda la operación de taladrado, hecho que no se produce en el caso sin control PID, en el cual la fuerza de corte se incrementa en el tiempo, a medida que aumenta la profundidad durante el corte.



**Figura 5.5. Comportamiento en tiempo real del sistema de control.**  
a) Fuerza de corte y b) velocidad de avance.

**Tabla 5.1. Criterios de error de la respuesta temporal de la fuerza de corte.**

Condición de operación	Criterios de error		
	<i>IAE</i>	<i>ITAE</i>	<i>ITSE</i>
Sin control PID	23.929	80.764	21.152
Control activado y filtro activado	19.552	32.86	6.818
Control PID	17.817	28.074	14.081

Se ha evaluado el error a estado estacionario de la fuerza de corte respecto a su valor de referencia, a través de tres criterios de error o índices de desempeño, muy ampliamente utilizados como son: la integral del error absoluto (*IAE*), la integral del error absoluto ponderado en el tiempo (*ITAE*) y la integral del error cuadrático ponderado en el tiempo

(ITSE). En la Tabla 5.1 se muestran los índices de error calculados para cada una de las condiciones de operación. Los valores de los índices relacionados en la tabla, reflejan la incuestionable superioridad del desempeño cuando se utiliza el sistema de control a lazo cerrado, encontrándose los índices entre un 32 y un 82 %, es decir entre 1.2 y 2.9 veces menor, que los índices de comportamiento del sistema a lazo abierto sin el controlador PID.

Por otra parte, el tiempo de la operación de mecanizado es menor cuando se utiliza el control a lazo cerrado. Al taladrar sin control activo, la velocidad de avance se programa en el CNC a un valor constante, el cual se escoge de forma conservativa, de tal manera que no se produzcan roturas de la herramienta durante el mecanizado. En el experimento realizado, el valor máximo de la fuerza de corte debe ser 1000 N, por lo que para que no se exceda este valor, la velocidad de avance programada debería ser entre 86 y 87 mm/min. De utilizar esta condición nominal, el tiempo de la operación de taladrado sería aproximadamente 1.3 % mayor que el tiempo de la operación con control a lazo cerrado, lo cual representa una ganancia importante en el tiempo de producción. Más aún, una ventaja tecnológica adicional como consecuencia de mantener prácticamente constante la fuerza de corte durante la operación de taladrado, es un aumento de la vida útil de la herramienta de corte, tal y como ha sido reportado anteriormente en la literatura por Oh et ál. [161].

## 5.2 Diseño e implementación de un sistema de inferencia neuro-borrosa para el control de un proceso de fabricación

En esta sección se presenta el diseño e implementación de un sistema de control, basado en la estrategia de control por modelo interno (CMI), para la regulación, también, de la fuerza de corte en el proceso de taladrado de alto rendimiento [234]. La estrategia CMI (ver Figura 5.6) implementa tanto para el modelo directo como para el inverso, un sistema de inferencia neuro-borrosa de tipo ANFIS (*Adaptive Network based Fuzzy Inference System*). El objetivo de control es mantener la fuerza de corte constante en toda la operación (variable controlada), a pesar de la presencia de perturbaciones debido al incremento de la profundidad de corte, modificando la velocidad de avance de la herramienta de corte (acción de control).

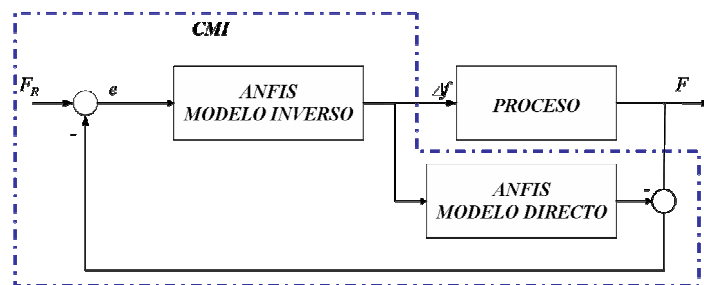


Figura 5.6. Estrategia CMI y modelos ANFIS.

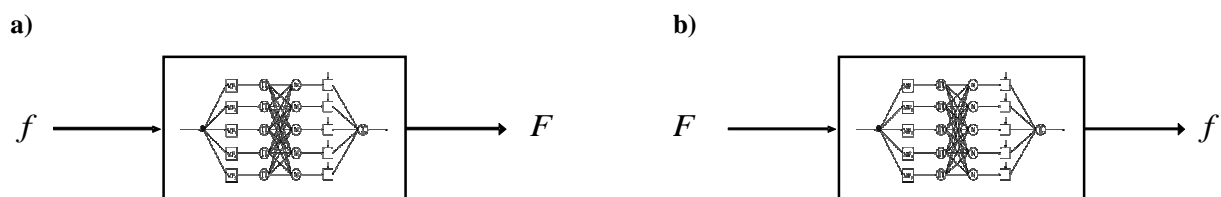
Los sistemas de inferencia neuro-borrosa, y particularmente ANFIS, resultan muy adecuados tanto para el modelado como para el control de procesos complejos, no-lineales y con un alto nivel de incertidumbre, como es el caso del proceso de taladrado de alto rendimiento. Por lo que el diseño de modelos de tipo ANFIS del proceso de TAR, con vista a ser implementados en una estrategia de control por modelo interno, puede resultar una opción muy interesante para controlar el proceso de forma adecuada.

Por otra parte, la decisión de utilizar también ANFIS para el modelado de la dinámica inversa del proceso está dada porque, en general, la inversión de modelos no lineales no es una tarea fácil y pueden no existir soluciones analíticas, de modo que las soluciones deben calcularse de forma numérica. Además, la inversión de modelos puede conducir a controladores inestables cuando el sistema es de fase no mínima [235].

En el apartado siguiente se obtendrán los modelos directo e inverso, como parte de la etapa de diseño del sistema de control. Seguidamente se simulará el comportamiento del sistema de control, se implementará en tiempo real y se realizará una comparación con otras estrategias de control aplicadas al proceso TAR.

### 5.2.1 Modelos directo e inverso

Teniendo en cuenta el modelo dinámico del proceso de TAR introducido en el Capítulo 2, se ha decidido que los modelos directo e inverso sean una representación simple entrada-simple salida del proceso. Para el caso del modelo directo (Figura 5.7.a), la velocidad de avance de la herramienta de corte ( $f$ ) es la variable de entrada y la fuerza media de corte ( $F$ ) la variable de salida. De la misma forma, en el caso del modelo inverso (Figura 5.7.b), este nos indica qué velocidad de avance se obtiene (salida) cuando existe una determinada fuerza de corte (entrada).



**Figura 5.7. Entradas y salidas de los modelos ANFIS.**  
a) Modelo directo y b) modelo inverso.

Una vez definida la estructura de los modelos, ambos sistemas neuro-borroso ANFIS, se entrenan para que aprendan el comportamiento a régimen transitorio del proceso de TAR controlado por alguna estrategia de control diseñada previamente. Para este caso, se emplearán datos experimentales entrada-salida de operaciones de taladrado controladas

mediante un sistema de inferencia borrosa de 9 reglas en su base de conocimiento y funciones de pertenencia de tipo trapezoidales. Este sistema fue ajustado mediante el método iterativo de prueba y error en pruebas de simulación. La estrategia de control, introducida por Haber et ál. [236], ha sido diseñada para el proceso TAR con el objetivo de controlar la fuerza de corte, variando la velocidad de avance de la herramienta y opera bajo las mismas condiciones de operación nominales que serán utilizadas por la estrategia CMI.

El conjunto de datos reales de fuerza de corte y de velocidad de avance se obtuvo de experimentos realizados en probetas de material de fundición nodular GGG40, con taladros de profundidad de hasta 15 mm. Las condiciones nominales de operación fueron: velocidad de giro de 870 rpm, velocidad de avance 100 mm/min y diámetro de la herramienta de 10 mm.

Con el objetivo de realizar un ajuste adecuado de ambos modelos, se ajustaron previamente a partir de un conjunto de datos de entrenamiento (3 experimentos con 133 muestras cada uno). Posteriormente, se ajustaron los parámetros de ambos modelos, mediante un conjunto de datos de prueba (2 experimentos con 128 muestras cada uno) distintos de los anteriores.

En la fase de obtención de los modelos son modificables ciertos parámetros del sistema ANFIS, tales como el número de funciones de pertenencia en el emborronado, tipo de dichas funciones, así como el orden de la función del consecuente de las reglas de la base de conocimiento o funciones de salida. Asimismo, se puede mejorar la precisión del modelo cambiando los parámetros del proceso de aprendizaje (método, número de iteraciones, tamaño del paso, etc.). La elección de las variables correctas y los parámetros óptimos se ha realizado a partir de un compromiso entre el criterio de error de la raíz cuadrada del error cuadrático medio (RMSE) y la respuesta del modelo.

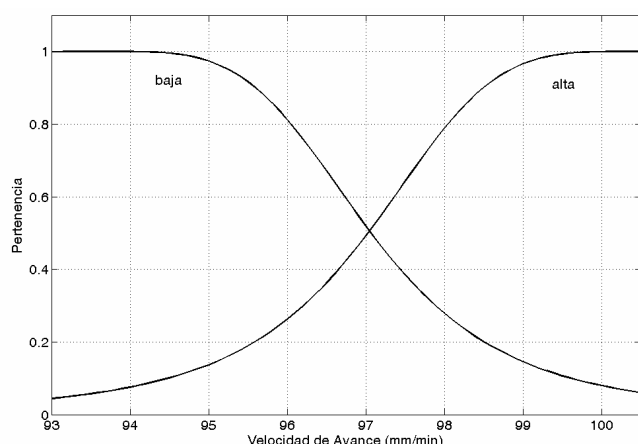
Se tuvieron en cuenta durante el diseño de los modelos, funciones polinómicas para los consecuentes de las reglas de orden cero y de primer orden, con funciones de pertenencia de tipo campana generalizada, sigmoidales, triangulares y trapezoidales, desde dos hasta nueve funciones de pertenencia. El mejor resultado para los modelos directo e inverso (ver Figura 5.8) se obtuvo con dos funciones de pertenencia en la fase de emborronado, siendo dichas funciones del tipo campana generalizada (extensión de la función de distribución de Cauchy-Lorentz) (ecuación (3.67)) y función del consecuente de las reglas polinomios de primer orden. El aumento del número de funciones de pertenencia y del orden de las funciones de salida no produjo mejoras significativas en la precisión.

Para el entrenamiento del modelo se han empleado funciones del paquete de herramientas de lógica borrosa de Matlab (*Fuzzy Logic Toolbox*). Los parámetros de entrenamiento fueron: 100 iteraciones del algoritmo (un mayor número de iteraciones provoca un sobreentrenamiento y, como resultado de éste, picos indeseados en la salida del sistema), modo de entrenamiento híbrido (únicamente con gradiente descendiente del error no se



alcanza el valor de salida deseado) y tamaño del paso de 0.01 (el aumento de este valor no produce una mejora significativa de la salida y si un mayor cómputo de operaciones). El tiempo de entrenamiento en los modelos, tanto para el modelo directo como para el inverso, fue de 0.14 segundos. Estos tiempos se obtuvieron en un ordenador con procesador Intel Core2 CPU 6400 - 2.13 GHz con sistema operativo Windows XP Profesional que no se encontraba conectado en línea con el proceso de TAR.

#### a) MODELO DIRECTO



##### Base de reglas:

- Regla 1: *SI* (*f* es “baja”) *ENTONCES* (*F* es “baja”) (1)  
Regla 2: *SI* (*f* es “alta”) *ENTONCES* (*F* es “alta”) (1)

##### Funciones de pertenencia de la entrada:

$$\mu(f) = \left[ 1 + \left( (f - c) / a \right)^{2b} \right]^{-1}$$

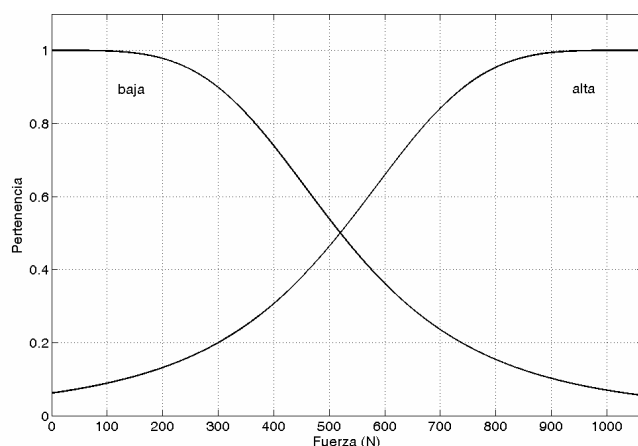
	<i>a</i>	<i>b</i>	<i>c</i>
Veloc. avance baja:	3.475	2.001	93.59
Veloc. avance alta:	3.473	1.998	100.5

##### Funciones del consecuente de cada regla:

Fuerza media baja (N):  $F = -62.3f + 6.8 \cdot 10^3$

Fuerza media alta (N):  $F = -322.9f + 3.2 \cdot 10^4$

#### b) MODELO INVERSO



##### Base de reglas:

- Regla 1: *SI* (*F* es “baja”) *ENTONCES* (*f* es “baja”) (1)  
Regla 2: *SI* (*F* es “alta”) *ENTONCES* (*f* es “alta”) (1)

##### Funciones de pertenencia de la entrada:

$$\mu(F) = \left[ 1 + \left( (F - c) / a \right)^{2b} \right]^{-1}$$

	<i>a</i>	<i>b</i>	<i>c</i>
Fuerza media baja:	520	1.983	$2.70 \cdot 10^{-4}$
Fuerza media alta:	520	1.964	1040

##### Funciones del consecuente de cada regla:

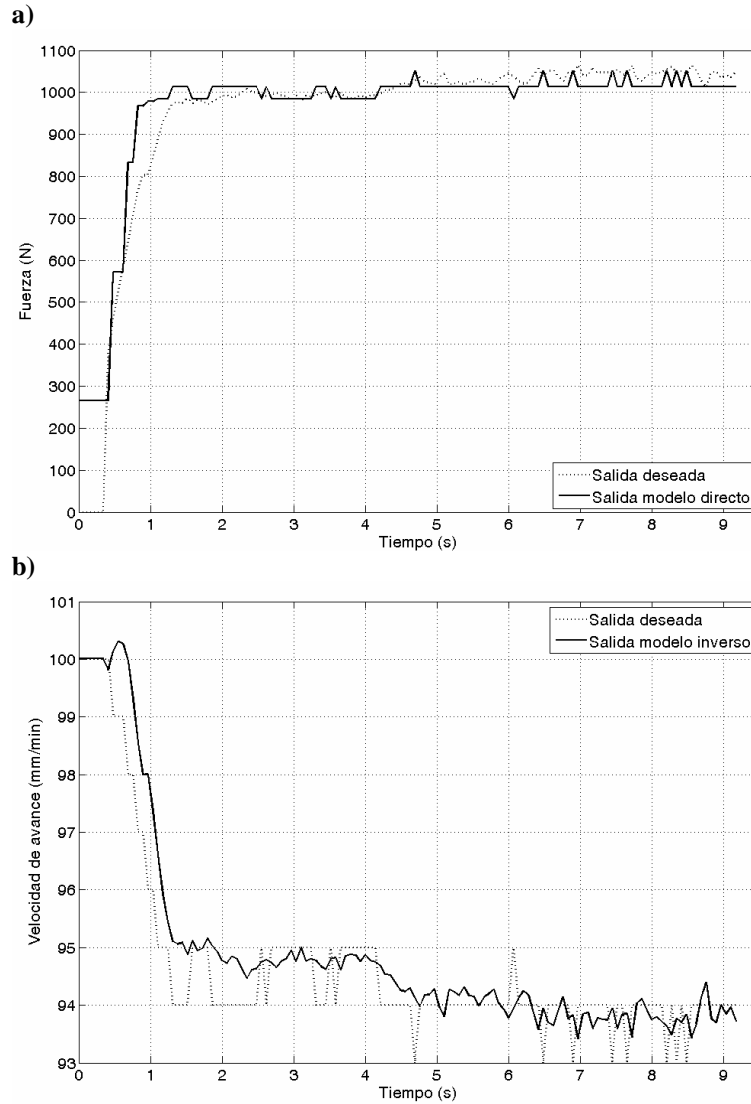
Veloc. avance baja (mm/min):  $f = -0.005F + 99.2$

Veloc. avance alta (mm/min):  $f = -0.02F + 113.9$

**Figura 5.8. Funciones de pertenencia y reglas de los modelos ANFIS.**

**a) Modelo directo y b) modelo inverso.**

En la Figura 5.9 puede observarse el desempeño de ambos modelos utilizando datos de entrada de validación, pudiéndose comparar con los datos reales del comportamiento del proceso (salida deseada de los modelos) obtenidos experimentalmente.



**Figura 5.9. Respuesta de los modelos ante datos de validación.**  
a) Modelo directo y b) modelo inverso.

Para validar numéricamente el desempeño de los modelos, se ha calculado el error medio cuadrático relativo (RMSE) entre su salida estimada y los datos experimentales, cuyos resultados se resumen la Tabla 5.2 junto con el resto de los parámetros de ambos modelos.

El error RMSE se calcula como:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N [(Y_i - F_i) / Y_i]^2} \quad (5.5)$$

donde  $F_i$  es valor obtenido por el modelo,  $Y_i$  es el valor deseado o real y  $N$  es el número total de muestras del experimento.

Tanto en el caso del modelo directo como en el caso del modelo inverso, se eligieron modelos que, a priori, tienen un error medio cuadrático mayor que otros modelos (con un

número diferente de funciones de pertenencia y de distinto tipo). Esta elección se hizo teniendo en cuenta el comportamiento a régimen transitorio (respuestas sin oscilaciones) y la sencillez del modelo para su implementación en un sistema de control en tiempo real del proceso de TAR. Un requisito fundamental del modelo directo es que su respuesta transitoria debe ser buena por la influencia negativa que tiene el máximo sobrepaso en la vida útil de la herramienta.

**Tabla 5.2. Resumen de los parámetros de los modelos neuro-borrosos.**

Parámetros	Modelo Directo	Modelo Inverso
Entrada-Salida	Simple-entrada / Simple-salida	Simple-entrada / Simple-salida
Tipo de función de pertenencia	Campana generalizada	Campana generalizada
Número de funciones de pertenencia	2	2
Tipo de Inferencia	Takagi-Sugeno	Takagi-Sugeno
Número de reglas	2	2
Iteraciones	100	100
Ratio de aprendizaje	0.01	0.01
Algoritmo de entrenamiento	Gradiente descendiente y mínimos cuadrados	Gradiente descendiente y mínimos cuadrados
Tiempo de entrenamiento	134.4 ms	134.4 ms
Tiempo medio de cómputo	0.01 ms	0.01 ms
<i>RMSE</i> - datos de entrenamiento	1.94 %	0.39 %
<i>RMSE</i> - datos de validación	5.18 %	0.68 %

### 5.2.2 Simulación del sistema de control

Después de haber obtenido los modelos directo e inverso ANFIS, la estrategia CMI se ha verificado mediante simulaciones en Matlab/Simulink, empleando el modelo dinámico de tercer orden del proceso de taladrado de alto rendimiento, presentado en el Capítulo 2, con un tiempo de retardo mayor de 0.4 segundos. En la Figura 5.10 se ilustra la aplicación de simulación, en la que se ha añadido a la entrada del modelo directo un filtro pasa bajo (5.6) con sus parámetros ajustados a  $k_1 = 0.6$  y  $k_2 = 0.8$ .

$$F(z) = (1 - k_1) / (z - k_2) \quad (5.6)$$

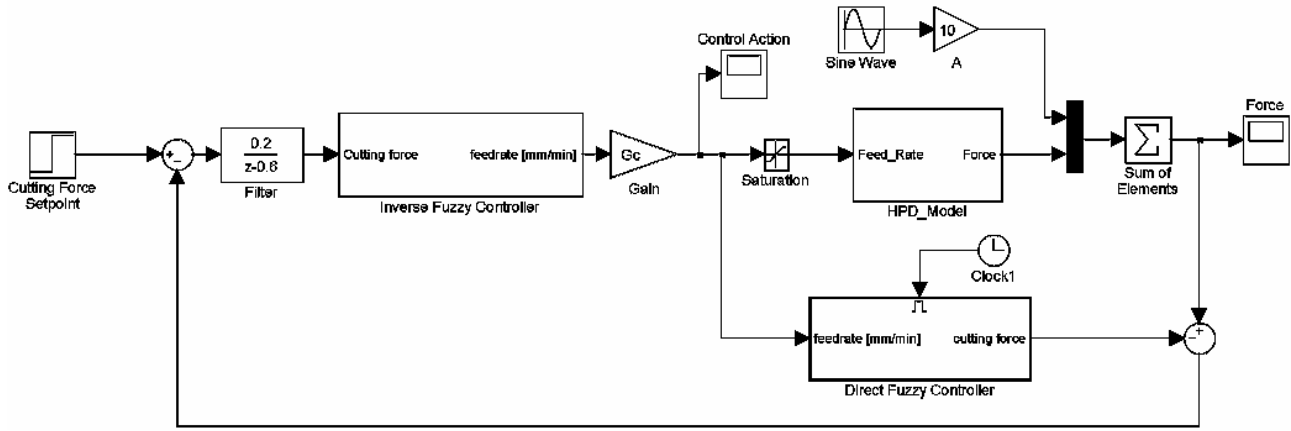


Figura 5.10. Aplicación de simulación de la estrategia de control por modelo interno y modelos neuroborrosos.

Asimismo, durante la simulación se incluyó una perturbación o ruido aditivo a la salida del proceso con el fin de estudiar la influencia de las dinámicas no modeladas en el sistema de control. La perturbación  $d(t)$  tiene la siguiente forma:

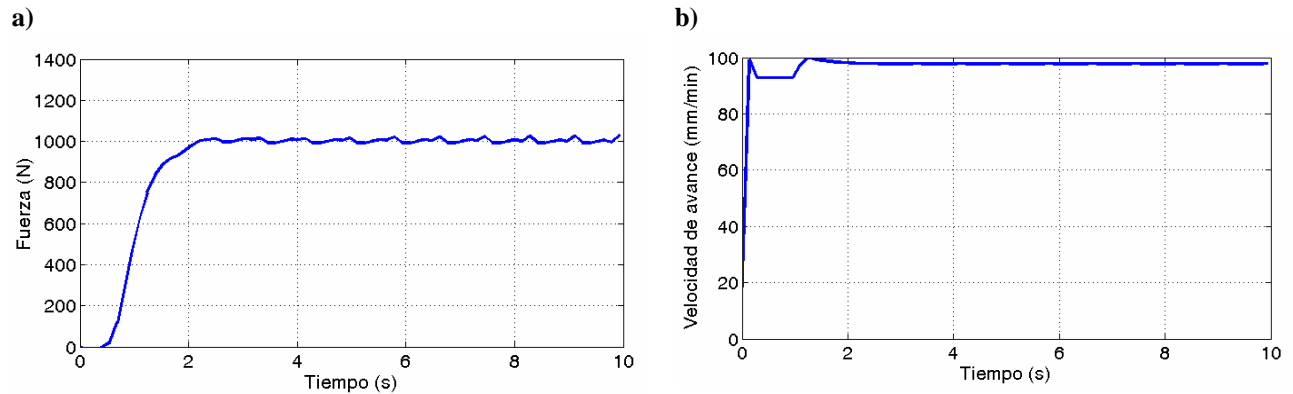
$$d(t) = A[\sin(2\omega t) + \sin(3\omega t) + \sin(4\omega t) + \sin(5\omega t)] \quad (5.7)$$

siendo  $\omega = 7.61$  rad/s. Esta frecuencia se corresponde con uno de los polos del modelo de tercer orden que representa de manera aproximada el proceso de taladrado. La amplitud de la perturbación se estima en  $A = 10$  para emular un ruido en la señal del 1 % de su nivel a estado estacionario (nivel de ruido bastante común en procesos de fabricación).

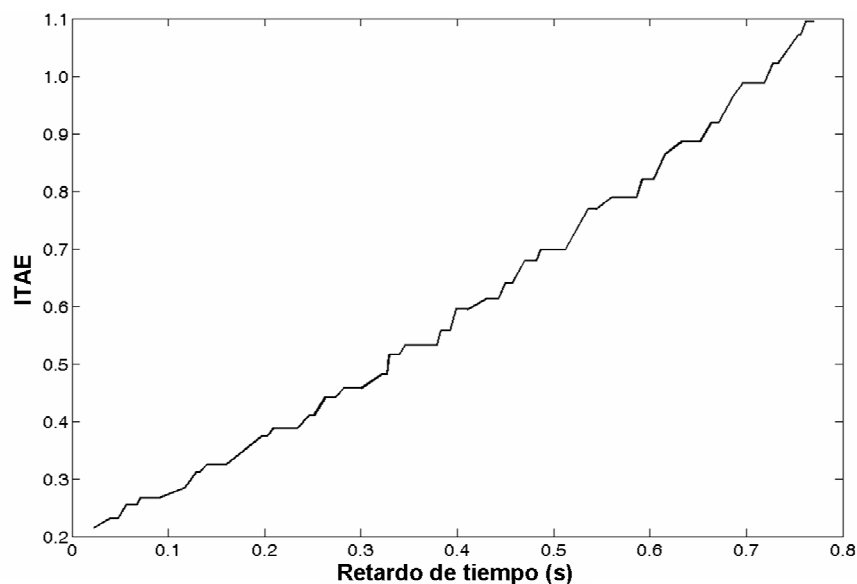
Se ha simulado una operación de taladrado de 15 mm (parámetro que se utilizará durante la validación del sistema de control en tiempo real), estimándose, previo a la simulación, una duración aproximada de la operación de 10 segundos y fijándose este tiempo como tiempo de simulación. Por otra parte, se ajustó como valor de referencia de la fuerza de corte el valor  $F_r = 1000$  N. Cabe recordar que el objetivo de control es mantener la fuerza de corte constante alrededor de su valor de referencia, una respuesta transitoria sin sobrepaso (amortiguamiento crítico) y bajo tiempo de establecimiento.

La Figura 5.11 muestra la respuesta del sistema de control por modelo interno del proceso de taladrado, en presencia de perturbación (5.7). Si analizamos la respuesta transitoria del sistema a lazo cerrado se puede comprobar que el máximo sobrepaso es del 2.36 %, y el tiempo de establecimiento es muy reducido (menor a 2 segundos, despreciando el retardo inicial). Por otra parte a partir de la respuesta temporal del error, se ha determinado el valor de

la integral del error absoluto ponderado en el tiempo,  $ITAE = 0.54$ , la integral del error absoluto,  $IAE = 0.96$  y la integral del error cuadrático ponderado en el tiempo,  $ITSE = 0.32$ .



**Figura 5.11. Simulación de la respuesta del sistema de control ante un escalón en la referencia**  
a) Fuerza de corte y b) acción de control.



**Figura 5.12. Comportamiento del criterio  $ITAE$  en presencia de retardos de tiempo.**

Por otra parte, se analizó el comportamiento del sistema de control en presencia de retardo de tiempo variable en el bus campo por el que se envía la acción de control al proceso de taladrado, realizándose simulaciones con diferentes retardos comprendidos entre  $[0.025, 0.75]$  segundos. En la Figura 5.12 se describe el comportamiento del índice de desempeño  $ITAE$  según varía el retardo de tiempo del proceso. El valor mínimo del criterio  $ITAE$  ha sido alcanzado por el retardo mínimo y viceversa.

### 5.2.3 Implementación experimental del sistema de control y estudio comparativo

Se realizaron pruebas experimentales de taladrado en la plataforma experimental. Se utilizó una herramienta de corte modelo R840-1000-30-A0A de Sandvik de 10 mm de diámetro y fundición nodular GGG40 como material de la pieza de trabajo, con una dureza Brinell de 233 HB. La velocidad de avance y la velocidad de giro nominal del cabezal de la máquina se ajustaron a  $f_0 = 100 \text{ mm/min}$  y  $n_0 = 870 \text{ rpm}$ , respectivamente. La profundidad máxima de la operación de taladrado es de 15 mm y el valor de referencia de la fuerza de corte es de 1000 N.

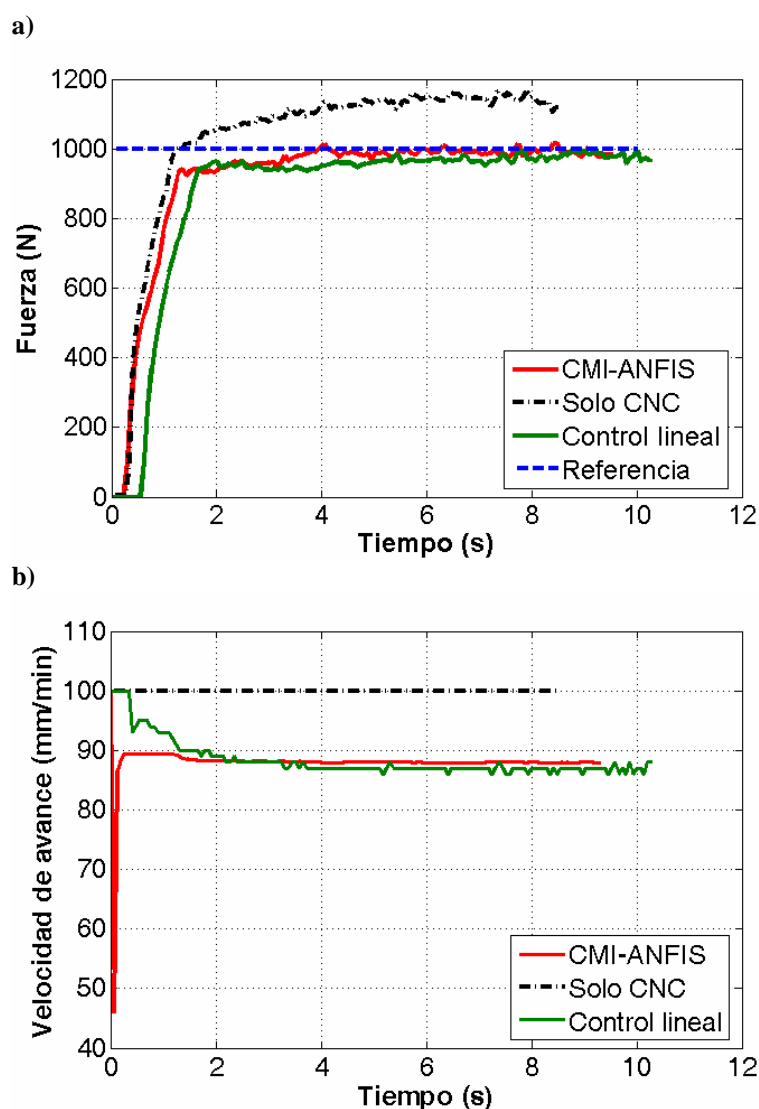


Figura 5.13. Comportamiento del sistema de control CMI-ANFIS y otras estrategias de control  
a) Fuerza de corte y b) velocidad de avance de referencia.

La estrategia de control se implementó en Matlab/Simulink sobre Windows, utilizando como interfaz de tiempo real la herramienta Real-Time Workshop (RTW) y enviando la acción de control a través del bus MPI a una frecuencia de 14.5 Hz. La señal de fuerza de corte es adquirida mediante el sistema de adquisición de datos, a una frecuencia de muestreo de 50 kHz.

En la Figura 5.13 se muestra la respuesta transitoria del sistema de control, representada por la variable controlada (fuerza de corte) y la velocidad de avance de referencia como acción de control. Como puede observarse, el sistema de control garantiza una respuesta temporal del proceso críticamente amortiguada (libre de oscilaciones), a pesar de las fuerzas de perturbación que surgen con el incremento de la profundidad de corte.

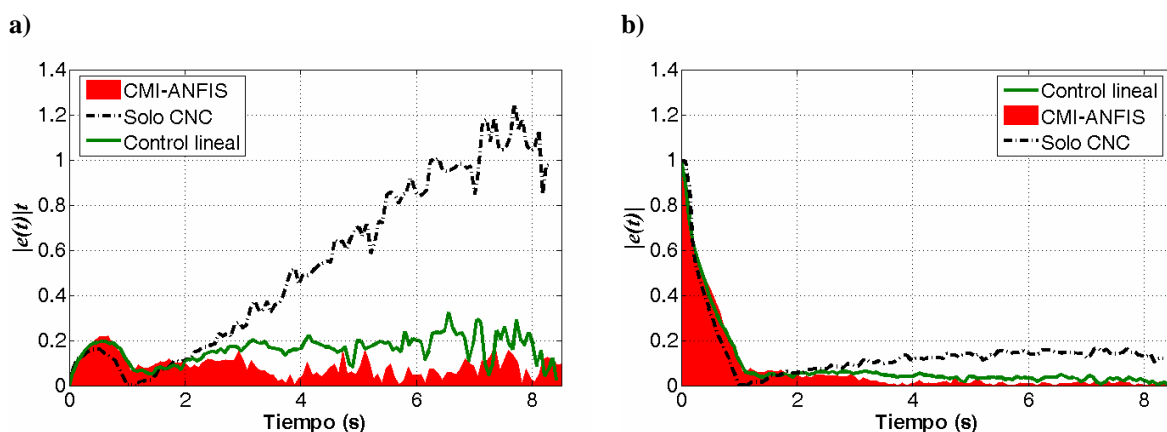
Para corroborar el buen desempeño del controlador, se ha realizado un estudio comparativo con otras estrategias de control, diseñadas para el mismo proceso y bajo las mismas condiciones de operación, cuyas respuestas temporales son mostradas también en la Figura 5.13. Por una parte el sistema CMI-ANFIS se compara con la estrategia de control lineal PID, ajustada mediante el método de Ziegler-Nichols, que fue introducida en este capítulo. Por otra parte, también se realiza una comparación con el comportamiento del proceso de taladrado de alto rendimiento a lazo abierto, es decir, controlado solamente mediante el CNC sin referencia externa de velocidad de avance.

**Tabla 5.3. Estudio comparativo entre el controlador CMI-ANFIS y otras estrategias de control analizadas.**

Controlador	Parámetros	ITAE	ITSE	IAE	Ovt (%)
CNC a lazo abierto	$f_0 = 100 \text{ mm/min}$ $n_0 = 870 \text{ rpm}$	4.737	0.722	1.387	16.363
PID Z-N, filtro en la referencia	$K_p = 0.01086$ $K_i = 0.01348$ $K_d = 0.00219$ $\tau = 0.95$	2.079	0.409	1.338	0.0
Controlador CMI-ANFIS	$k_1 = 0.6$ $k_2 = 0.8$ $G_c = 1.0$	1.032	0.168	0.83	1.549

La Tabla 5.3 muestra los índices de error calculados a partir de la respuesta temporal de las estrategias de control implementadas en tiempo real. Para el cálculo se ha utilizado como tiempo inicial ( $t_0 = 0$ ) el instante cuando se produce el contacto entre la herramienta y el material de trabajo. La estrategia CMI-ANFIS tiene un mejor desempeño que el resto de estrategias bajo los criterios *ITAE*, *IAE* e *ITSE*. Esta misma conclusión puede corroborarse a

partir de las gráficas de la Figura 5.14, donde se describe, para cada una de las estrategias de control, el error absoluto ponderado en el tiempo  $|e(t)|t$  (Figura 5.14.a) y el error absoluto  $|e(t)|$  (Figura 5.14.b) durante toda la operación de taladrado. El error está calculado relativo al valor de referencia de la fuerza de corte. En cada una de las gráficas aparece sombreada el área debajo de la curva correspondiente al sistema de control CMI-ANFIS y que representa el valor de los índices *ITAE* e *IAE*, respectivamente.



**Figura 5.14. Comportamiento del error relativo para las estrategias de control**  
a) Error absoluto ponderado en el tiempo y b) error absoluto.

El buen comportamiento de la estrategia CMI-ANFIS se verifica tanto al principio como al final de la operación: respuesta suave al principio y en torno al valor de referencia a lo largo de todo el proceso de taladrado. Esto se traduce en que la entrada de la herramienta de corte en la pieza es mucho más suave y en que la evacuación de material en la parte final se hace de un modo más correcto (factores importantes para reducir el desgaste de herramienta y mejorar la calidad superficial).

El controlador CMI-ANFIS propuesto es sencillo y fácil de implementar, previa existencia de datos entrada/salida del proceso. Para este caso de estudio, solo cuatro reglas (dos para el modelo directo y dos para el inverso) han resultado suficientes para implementar la ley de control y asegurar el comportamiento deseado.

### 5.3 Diseño e implementación de un sistema de inferencia borrosa aplicado al control del proceso de taladrado de alto rendimiento. Optimización mediante el método de Nelder-Mead

En esta sección se presenta el diseño e implementación de un sistema de control basado en un sistema de inferencia borrosa. Se realiza un estudio, análisis y verificación, del



comportamiento de un sistema de control borroso de 49 reglas, de dos entradas y una salida, durante el control, en una arquitectura distribuida, de un proceso complejo como el taladrado de alto rendimiento (TAR). En la etapa de diseño y ajuste del sistema de control, se emplea un algoritmo numérico sencillo y computacional eficiente para minimizar una función objetivo, dentro de un espacio multidimensional, representada por un índice de desempeño que penaliza respuestas transitorias de larga duración del sistema de control y con pequeños sobrepasos y oscilaciones [221, 237, 238]. Para el proceso de optimización se ha utilizado, inicialmente, el método de Nelder-Mead y como función objetivo la integral del error absoluto ponderado en el tiempo (*ITAE*).

La idea de emplear un algoritmo de control no-lineal como el sistema borroso, surge por el evidente carácter no-lineal de la fuerza de corte en el proceso de TAR a estado estacionario que, como se ha observado con anterioridad, sufre variaciones a medida que se incrementa la profundidad de corte en una operación de taladrado. En la operación de taladrado, el operador humano fija parámetros de operación conservadores constantes sobre la base de la experiencia pasada, con el fin de evitar un rápido desgaste de la herramienta o una rotura catastrófica de esta. Sin embargo, mediante el empleo de un controlador borroso, es posible ajustar la velocidad de avance con el objetivo de regular la fuerza de corte, optimizando de esta manera la tasa de producción sin que sea necesario estar pendientes de posibles daños en la herramienta de corte, debido a un posible sobre esfuerzo. En particular, el controlador borroso puede ser capaz de regular la fuerza de corte ante incrementos en la fricción entre la herramienta y la pieza de trabajo o ante cambios en la dureza del material de trabajo.

### 5.3.1 Diseño de un controlador borroso para el proceso TAR

Un controlador borroso muy habitual en la industria y en la literatura, describe la relación entre el cambio de la señal de control (por ejemplo el cambio  $\Delta f$  en la velocidad de avance en el taladrado) y la señal de error del sistema de control a lazo cerrado (por ejemplo, error  $e_F$  en la fuerza de corte y su variación  $\Delta e_F$ ) [239]. Este tipo de controlador fue sugerido a principio de los años 70 y se ha convertido en un estándar en el campo del control borroso [240]. Para el diseño del controlador de dos entradas y una salida se han seguido pasos perfectamente establecidos, como son: la definición de las funciones de pertenencia de las variables de entrada y de la salida y la construcción de las reglas de control difusas, que forman la base de conocimiento del controlador. El controlador borroso realiza acciones de control en línea, modificando la velocidad de avance de referencia del CNC de la máquina herramienta, resultando entonces como variable manipulada, el incremento en la velocidad de avance alrededor del valor nominal programado en el CNC.

El valor de referencia de la fuerza de corte  $F_r$ , se obtiene de una fase de aprendizaje, a partir de los experimentos de identificación del proceso de taladrado de alto rendimiento. En cada período de muestreo, el error en la fuerza de corte  $e_F$  y su variación  $\Delta e_F$ , ambos en Newton (N), se calculan como:

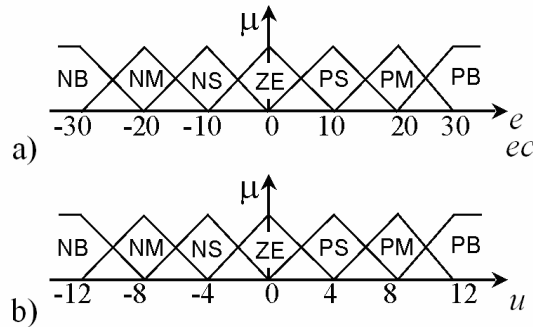
$$e_F(k) = F_r(k) - F(k) \quad (5.8)$$

$$\Delta e_F(k) = e_F(k) - e_F(k-1) \quad (5.9)$$

La salida del controlador  $u$  y el vector de error de entrada se determinan, respectivamente, como:

$$[e \ ec] = \begin{bmatrix} K_e \cdot e_F & K_{ce} \cdot \Delta e_F \end{bmatrix}; u = \Delta f \quad (5.10)$$

siendo  $e$  y  $ec$  las variables discretas de entrada y  $K_e$ ,  $K_{ce}$  factores de escala de cada una de las entradas, respectivamente.



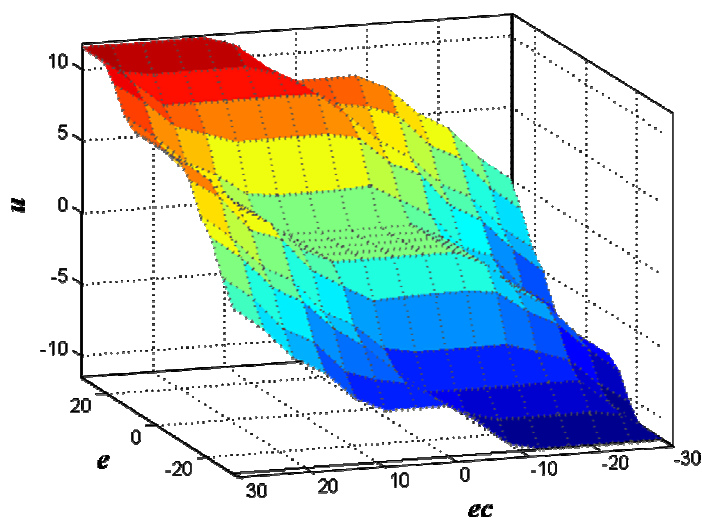
**Figura 5.15. Particiones borrosas y funciones de pertenencia de las entradas y salidas.**  
a)  $e$ ,  $ec$  y b) salida  $u$ .

Por otra parte, el universo de discurso de las variables de entrada se ha dividido en 7 conjuntos difusos, caracterizados en cada intervalo por funciones de pertenencia de forma triangular equidistantes dentro del rango  $[-30, 30]$ . Para la variable de salida se definen también siete particiones equidistantes con funciones de pertenencia triangulares. El universo de discurso de esta variable se establece de acuerdo a la máxima modificación posible (alrededor de un 12%) en el valor nominal de la velocidad de avance, que se determina a partir de las condiciones nominales de la operación de corte. La Figura 5.15 muestra las particiones borrosas resultantes para la salida  $u$  y el vector de error  $[e \ ec]$ . Los siete conjuntos difusos utilizados para las entradas y la salida son: NB muy negativo, NM negativo medio, NS negativo pequeño, ZE cero, PS positivo pequeño, PM positivo medio y PB positivo grande.

La selección de la forma de las funciones de pertenencia es el resultado de la combinación de un procedimiento de prueba y error con un criterio de diseño empírico, establecido sobre la base del conocimiento ingenieril y de estudios de simulación. Las funciones de pertenencia de forma triangular, son las de uso más frecuente en controladores borrosos debido a su eficiencia computacional y fácil implementación en aplicaciones en tiempo real.

**Tabla 5.4. Base de reglas del controlador borroso.**

$e \backslash ec$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS
NS	NB	NB	NM	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB



**Figura 5.16. Característica estática del controlador borroso.**

La construcción de la base de reglas se realiza a partir de una plantilla que es considerada como el sentido común con mayor coincidencia entre ingenieros con experiencia en control borroso. Se ha elegido para la creación de reglas de la base la propuesta de plantilla de

MacVicar-Whelan [241], modificando de forma apropiada las etiquetas lingüísticas de la variable de salida. La Tabla 5.4 resume las 49 reglas de control que conforman la base de conocimiento. Una base de reglas similar ha sido aplicada previamente al control borroso, en un entorno distribuido, de un sistema electromecánico [242]. Por otra parte, la característica estática resultante del controlador se describe en la Figura 5.16.

Como método composicional de inferencia se ha seleccionado el operador de composición “sup-producto”. Un controlador con este tipo de operador trabaja con una precisión razonable en diferentes aplicaciones [243]. Empleando la operación de producto cartesiano para cada implicación borrosa y utilizando el operador máximo para el cálculo de la acción de control borrosa resultante, se tiene:

$$\mu_R(e, ec, u) = \max_{i=1}^{49} \left\{ \text{prod} \left\{ \mu_{e_i}(e), \mu_{ec_i}(ec), \mu_{u_i}(u) \right\} \right\} \quad (5.11)$$

El valor determinista de la salida del controlador que se corresponde con el cambio en la velocidad de avance, se obtiene a partir de la etapa de *desemborronado*. Básicamente existen tres tipos de estrategias de *desemborronado* más comúnmente utilizadas: el máximo, la media del máximo (MDM) y el centro de área (CDA) [244]. Debido a un apropiado desempeño a estado estacionario o de equilibrio, además de su uso como método de *desemborronado* estándar en controladores borrosos, tanto experimentales como industriales, se ha escogido CDA como estrategia de *desemborronado*. Dados unos valores deterministas de entrada al controlador ( $e, ec$ ), su valor determinista de salida  $u$  se expresa como:

$$u = \frac{\sum_i \mu_R(u_i) \cdot u_i}{\sum_i \mu_R(u_i)} \quad (5.12)$$

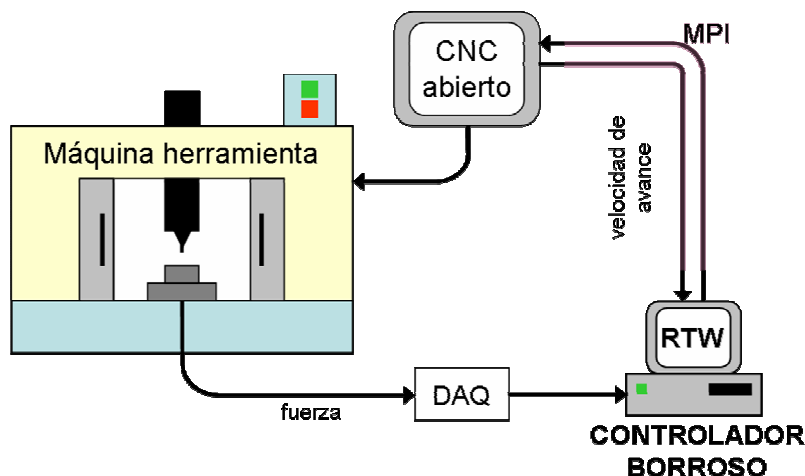
Finalmente, la acción de control final  $f$  (velocidad de avance) que se envía al CNC de la máquina herramienta, como valor de referencia de la velocidad de avance de la máquina, se calcula multiplicando el factor de escala de salida  $GC$  por la salida determinista del controlador borroso, lo cual se expresa como:

$$f(k) = f(k-1) + GC \cdot u(k) \quad (5.13)$$

### 5.3.2 Ajuste e implementación del controlador borroso

El controlador borroso de 49 reglas introducido anteriormente está insertado en la arquitectura de control distribuido del proceso de taladrado de alto rendimiento, que a su vez forma parte de la plataforma experimental introducida previamente. La variable a controlar es

la fuerza de corte durante la operación de taladrado y su acción de control es la velocidad de avance de referencia del CNC de arquitectura abierta. La fuerza de corte se adquiere mediante el sistema de adquisición de datos de la plataforma y la señal de acción de control es enviada a través del bus de campo MPI (ver Figura 5.17). El controlador se ha implementado en Matlab/Simulink sobre Windows, utilizando como interfaz de tiempo real la herramienta Real-Time Workshop (RTW).



**Figura 5.17. Sistema de control distribuido basado en un regulador borroso para el control del proceso de TAR**

Para el ajuste del control se ha utilizado como función objetivo a optimizar, el criterio o función de coste *ITAE*, que permite obtener una respuesta transitoria del sistema de control con pequeño tiempo de establecimiento y sin sobrepasos. Otros criterios a tener en cuenta serán la integral del error cuadrático ponderado en el tiempo (*ITSE*), la integral del error absoluto (*IAE*) y la integral del error cuadrático ponderado por el tiempo al cuadrado (*IT<sup>2</sup>SE* o *ISTSE*). En específico, el criterio *ITAE* penaliza las respuestas transitorias de larga duración y es muy sensible ante variaciones de parámetros en comparación con otros criterios. Por otra parte, el criterio *ITAE* por lo general produce menores sobrepasos y oscilaciones que los criterios *IAE* e *ITSE*.

El comportamiento del sistema de control ha sido examinado mediante simulación en Matlab/Simulink empleando, como aplicación, el diagrama ilustrado en la Figura 5.18. El criterio *ITAE* describiría el comportamiento del sistema ante perturbaciones externas. En este estudio se ha considerado como perturbación un cambio de tipo escalón en el valor de referencia de la fuerza de corte, y el criterio *ITAE* es una buena opción para evaluar cuan bien es capaz el sistema de control de seguir los cambios en la referencia.

El objetivo del procedimiento de optimización es minimizar el índice *ITAE*, ajustando las ganancias o factores de escala de entrada del controlador borroso,  $[K_e \ K_{ce}]$ . Los valores óptimos para estos parámetros se determinan como:

$$\mathbf{K}_{OPT} = [K_e \ K_{ce}]_{OPT} = \arg \min \left( \int_0^T |e_F(t)| t dt \right) \quad (5.14)$$

$$Ovt = 100(\max(e_F) - F_r) / F_r \quad (5.15)$$

donde  $T$  es el tiempo de duración de toda la operación de taladrado, considerando como tiempo inicial  $t = 0$  cuando la herramienta hace contacto con el material,  $e_F$  es la señal de error entre la fuerza de referencia  $F_r$  y la fuerza de corte medida y  $Ovt$  es el máximo sobrepaso de la respuesta dinámica a lazo cerrado.

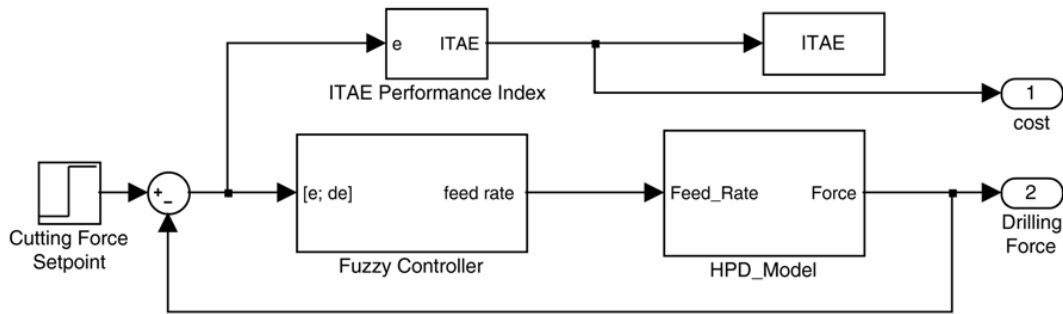


Figura 5.18. Aplicación de simulación del sistema de control borroso.

Valores altos de la ganancia  $K_e$  del controlador borroso significan una buena capacidad de respuesta del sistema de control, sintetizada en un error muy pequeño a estado estacionario y bajo tiempo de establecimiento. Sin embargo esto puede conllevar un empeoramiento de la estabilidad del sistema y provocar altos niveles de sobrepasos en su respuesta temporal. Por otra parte, una convergencia más rápida de la variable controlada hacia su valor de referencia, se obtiene con la ganancia  $K_{ce}$  pequeña. Existen varios métodos para ajustar controladores borrosos, pero no es objetivo de este trabajo hacer un estudio de estos en profundidad.

Los valores óptimos de las ganancias de entrada del controlador  $[K_e \ K_{ce}]$ , fueron determinados mediante el algoritmo de optimización de Nelder-Mead [221]. Este algoritmo es un método numérico para minimizar una función objetivo en un espacio multidimensional. El método utiliza el concepto de un *símplex*, que es una forma geométrica de  $M+1$  vértices, siendo  $M$  el número de parámetros a optimizar. El paso básico del método consiste en

sustituir el peor punto del simplex con otro punto reflejado en el resto de puntos, a través del cálculo del centro de gravedad de estos  $M-1$  puntos. Resulta típico el empleo de un simplex de tamaño constante y pequeño, que permita desplazarse mediante gradientes locales hasta los máximos o mínimos locales [221].

La función de coste fue evaluada mediante simulación del sistema de control, donde el proceso de TAR se representa por su modelo dinámico de tercer orden. Al ajustar los parámetros del controlador a los siguientes valores iniciales:  $[K_e \ K_{ce}]_{INIT} = [1.0, 1.0]$  y  $G_c = 0.5$ , se obtiene un máximo sobrepaso de 4.090 % y un índice  $ITAE$  igual a 0.709.

Para el procedimiento de optimización del sistema de control se utilizaron los siguientes parámetros de ajuste del algoritmo Nelder-Mead: 400 evaluaciones máximas de la función objetivo, 400 iteraciones máximas permitidas y  $1.00e-004$  como tolerancia tanto de la evaluación de la función objetivo como los parámetros de entrada de la función. Durante todo el procedimiento la ganancia o factor de escalado de salida del controlador borroso  $G_c$ , se mantuvo constante.

Tras ejecutar el algoritmo de optimización, se alcanzó un valor mínimo  $ITAE = 0.469$ , correspondiéndose a  $[K_e \ K_{ce}]_{OPT} = [0.0125, 0.0374]$ . Por otra parte se obtuvo un máximo sobrepaso  $Ovt = 0.15\%$ , un valor del índice  $ITSE$  de 0.292 e  $IT^2SE = 0.164$ .

Los resultados de la simulación del sistema de control son mostrados en la Figura 5.19. En ella se puede observar tanto el comportamiento de la fuerza de corte ante un cambio de tipo escalón en su valor de referencia, como en la señal de control representada por la velocidad de avance de referencia. Como cabría de esperar, se consigue una mejora clara en la respuesta transitoria del sistema tras el procedimiento de optimización.

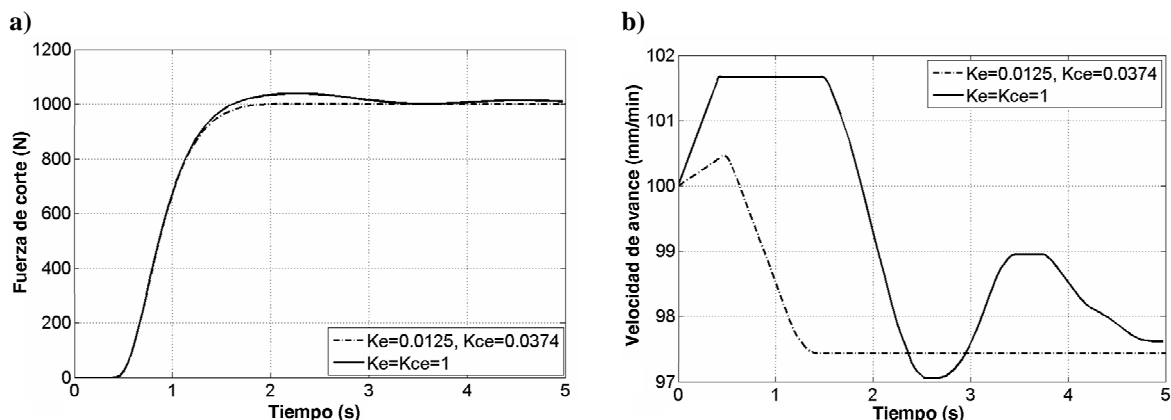


Figura 5.19. Resultados de la simulación del sistema de control.

a) Fuerza de corte y b) velocidad de avance.

Por otra parte, se ha realizado un estudio de la influencia de los retardos de las comunicaciones de la arquitectura distribuida en el sistema de control. En estudios anteriores se ha analizado, desde el punto de vista del sistema de control, las limitaciones que introducen las redes de comunicaciones. Se han estudiado los retardos de tiempo en un sistema de control distribuido sobre una red conmutada, empleando TCP / IP como protocolo de transmisión y con cierto tiempo de latencia. Por lo general, los retrasos han sido modelados empleando técnicas estadísticas tales como funciones de distribución de exponencial negativa y cadenas de Poisson y Markov [245].

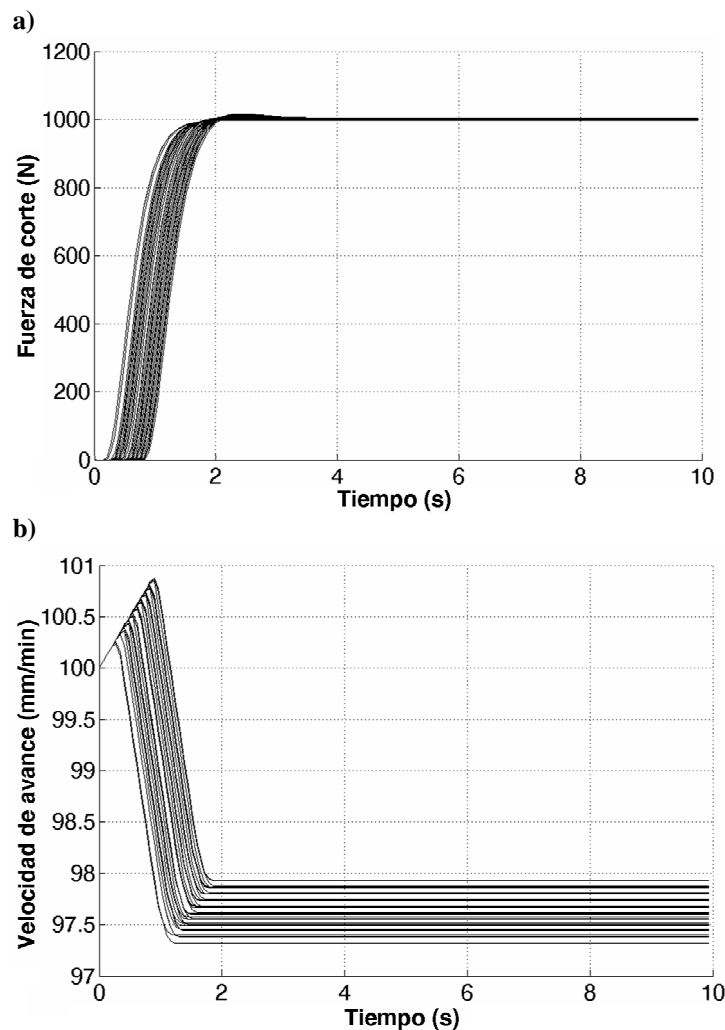


Figura 5.20. Comportamiento del sistema de control según varía el tiempo de retardo.  
a) Fuerza de corte y b) variaciones en la velocidad de avance.

Sin embargo, estos modelos tienen una validez limitada, debido al comportamiento no determinista de algunas redes de comunicación. Para estudiar la influencia del retardo de tiempo en el desempeño del controlador, se ha incluido en la simulación un simple generador de retardos. Se asume conocido el tiempo de retardo máximo y, con el objetivo de mantener



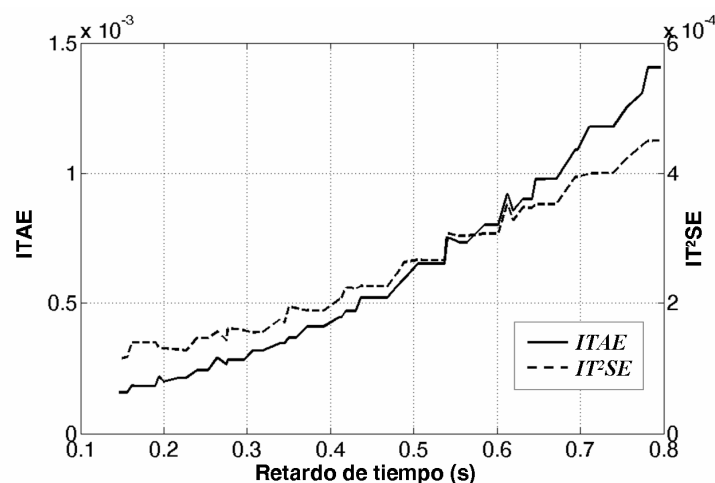
bajo el coste computacional, se ha seleccionado para el generador una función de distribución uniforme. El retraso de tiempo variable,  $L$ , fue simulado asumiendo un retraso uniformemente distribuido entre 0.138 s (el doble del tiempo de accionamiento del lazo) y 0.8 s (el doble del retardo del modelo), ejecutándose 100 pruebas de simulación para el sistema de control borroso optimizado. El máximo, el mínimo y la media de los valores que alcanza el criterio  $ITAE$ , además del máximo sobrepaso, son calculados a partir de estas simulaciones.

La Figura 5.20 muestra los resultados de la simulación correspondiente a la respuesta a escalón del sistema de control, para todos los retardos de tiempo, resumiéndose tanto el peor como el mejor caso (mayor y menor  $ITAE$ ) en la Tabla 5.5. A partir de los resultados de la simulación, se deduce que el ajuste óptimo realizado garantiza una respuesta transitoria sin sobrepaso con un tiempo de establecimiento de menos de 2 segundos.

**Tabla 5.5. Comportamientos extremos del sistema de control para el tiempo de retardo variable.**

$ITAE$	Retardo $L$ (s)	Máximo Sobrepaso (%)
0.288 (Mínimo)	0.138	0.0
1.125 (Máximo)	0.7962	1.701

La Figura 5.21 describe la dependencia de los índices de desempeño con el tiempo de retardo. El deterioro de los índices  $ITAE$  e  $IT^2SE$  aumenta de forma no lineal con el incremento del retardo. Aunque este deterioro en los dos índices era de esperarse, se demuestra la robustez del controlador borroso de 49 reglas en presencia de tiempo muerto y retardo de tiempo en el proceso controlado y en la red de comunicación.



**Figura 5.21. Comportamiento de los índices de desempeño  $ITAE$  e  $IT^2SE$  según varía el tiempo de retardo.**

### 5.3.3 Implementación experimental del sistema de control y estudio comparativo

Se realizaron pruebas experimentales de taladrado en la plataforma experimental. Se utilizó una herramienta de corte de Sandvik de 10 mm diámetro y el material de la pieza de trabajo fue GGG40 con una dureza Brinell de 233 HB. La velocidad de avance y la velocidad de giro nominal del husillo se ajustaron a  $f_0 = 100 \text{ mm/min}$  y  $n_0 = 870 \text{ rpm}$ , respectivamente, y se estableció en 14 mm la profundidad máxima de la operación de taladrado. El valor de referencia de la fuerza de corte se fijó en 1000 N.

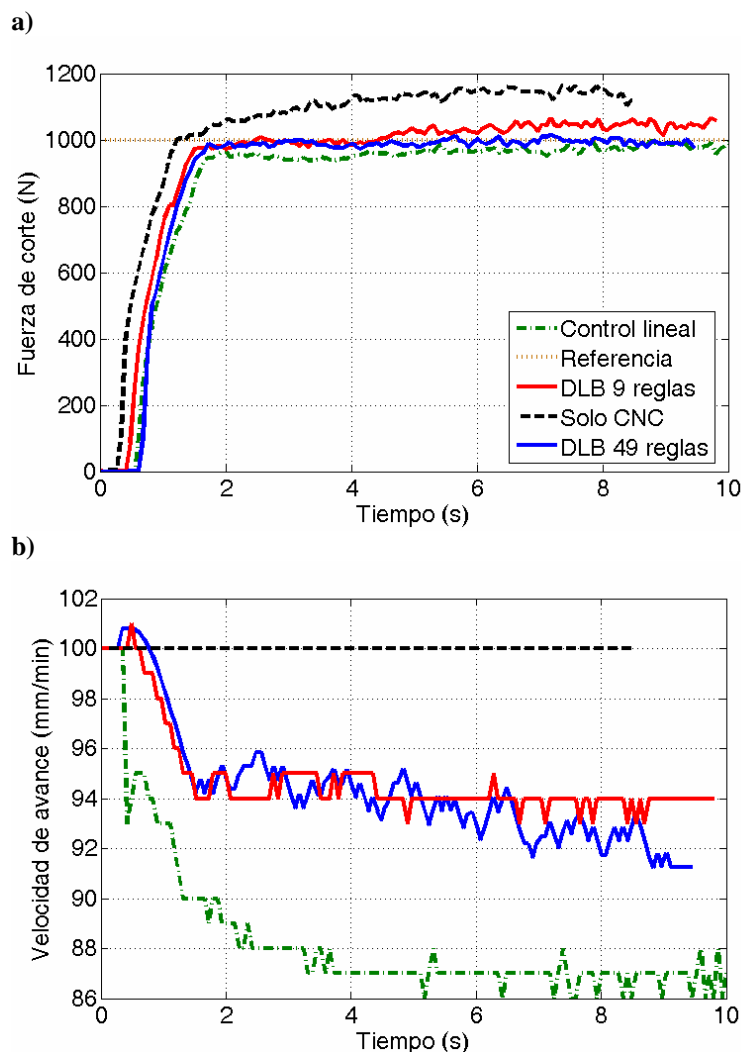


Figura 5.22. Comportamiento del proceso según las estrategias de control PID y borrosas.  
a) Fuerza de corte y b) velocidad de avance de referencia.

En la Figura 5.22 se describe la respuesta transitoria del sistema de control, representada por la variable controlada (fuerza de corte) y la velocidad de avance de referencia como acción de control. Como puede observarse, el sistema de control garantiza una respuesta

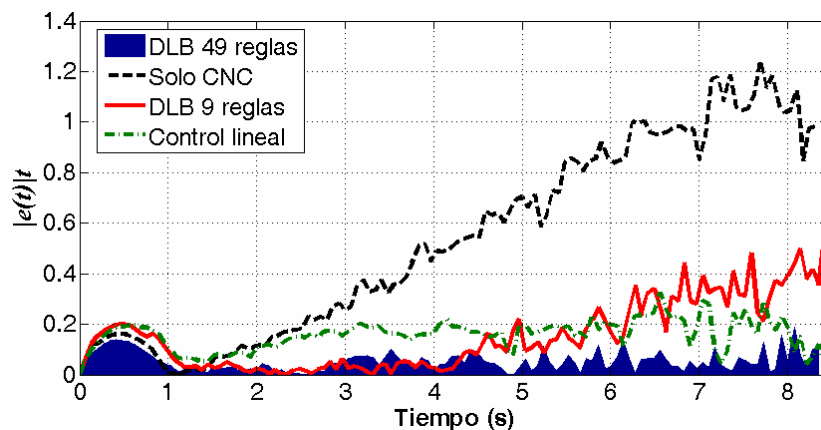
temporal del proceso libre de oscilaciones, a pesar de las fuerzas de perturbación que surgen con el incremento de la profundidad de corte.

Para corroborar el buen desempeño del controlador, se ha realizado un estudio comparativo con otras estrategias de control, diseñadas para el mismo proceso y bajo las mismas condiciones de operación, cuyas respuestas temporales son mostradas también en la Figura 5.22. En un trabajo previo, Haber et ál. [236] presentan el diseño e implementación de un controlador borroso de 9 reglas en su base de conocimiento, funciones de pertenencia de tipo trapezoidal y ajustado mediante el método iterativo de prueba y error en pruebas de simulación. En el estudio comparativo se ha tenido en cuenta, además, el controlador clásico PID con filtrado de referencia y optimizado mediante el método de Ziegler-Nichols, introducido en secciones anteriores.

**Tabla 5.6. Estudio comparativo de las estrategias de control analizadas.**

Controlador	Parámetros	<i>ITAE</i>	<i>ITSE</i>	<i>IAE</i>	<i>IT<sup>2</sup>SE</i>	<i>Ovt</i> (%)
CNC a lazo abierto	$f_0 = 100 \text{ mm/min}$ $n_0 = 870 \text{ rpm}$	4.737	0.722	1.387	3.874	16.363
PID Z-N	$K_p = 0.01086$ $K_i = 0.01348$ $K_d = 0.00219$ $\tau = 0.95$	2.079	0.409	1.338	0.5386	0.0
Controlador borroso de 9 reglas (Haber et ál., 2008)	$K_e = 0.0559$ $K_{ce} = 0.1156$ $GC = 1.0$	1.990	0.297	1.089	0.660	6.575
Controlador borroso de 49 reglas	$K_e = 0.0125$ $K_{ce} = 0.0374$ $GC = 0.5$	0.967	0.346	1.087	0.242	1.471

La Tabla 5.6 muestra los índices de error (relativo al valor de referencia de la fuerza) calculados a partir de la respuesta temporal de las estrategias de control implementadas en tiempo real. A primera vista, el comportamiento del controlador borroso de 49 reglas parece muy similar al resto. Sin embargo, índices como el *ITAE*, el *IT<sup>2</sup>SE* y el *IAE*, indican una mejora en la respuesta dinámica del proceso empleando la estrategia de control borrosa de 49 reglas en su base de conocimientos. Con vistas a realizar una mejor evaluación gráfica del desempeño temporal de las estrategias de control, la Figura 5.23 ilustra el comportamiento del error absoluto ponderado en el tiempo ( $|e(t)|t$ ) para cada una ellas. Se ha rellenado el área debajo de la curva correspondiente a la estrategia de control borrosa de 49 reglas. El valor de esta área representa el índice *ITAE* y como puede comprobarse es menor al área debajo del resto de curvas.



**Figura 5.23.** Comportamiento del error absoluto ponderado en el tiempo de las estrategias de control.

Los resultados experimentales confirman que es posible suprimir los aumentos y fluctuaciones en la fuerza de corte, durante la operación de taladrado, debido al incremento de la profundidad de corte, con una marcada mejora del proceso empleando el sistema de control borroso de 49 reglas. Los sobrepasos y oscilaciones de la fuerza de corte son perjudiciales para la vida útil de la herramienta, así como para la calidad de las operaciones de taladrado y por tanto deben ser evitados.

### 5.3.4 Estabilidad del sistema de control

El análisis de estabilidad de los sistemas de control distribuidos, no entra entre los objetivos de este trabajo. A pesar de que el control borroso fue inicialmente introducido como un método de control con un diseño basado en el conocimiento de un operador humano, la investigación realizada está basada casi exclusivamente en métodos de control borroso basados en modelos, que puede garantizar estabilidad y robustez del sistema de control a lazo cerrado [246]. Varios trabajos previos resaltan la necesidad de desarrollar métodos que garanticen la estabilidad global del sistema de control borroso desde la fase de diseño [247-249].

El análisis de estabilidad de un sistema de control borroso tiene ciertas limitaciones en cualquier problema de optimización, ya que los factores de escala pueden influir drásticamente en la dinámica de todo el sistema de control a lazo cerrado. Existen varios estudios acerca del empleo de métodos eficaces para el ajuste de factores de escala. Algunos de ellos sólo se enfocan en el ajuste del factor de escala de la salida del controlador borroso debido a su fuerte influencia en el desempeño y la estabilidad del sistema de control [250]. Otros trabajos sugieren que, si los parámetros del controlador borroso cumplen el Teorema de Pequeña Ganancia (*Small-Gain Theorem*), se satisface la estabilidad del sistema lazo cerrado

[251]. Sin embargo, si hubiera disponible una aproximación lineal del proceso, la cual caracteriza sus componentes dinámicas principales en al menos una vecindad del punto de operación, entonces resultaría factible realizar una prueba de estabilidad local y asintótica. La realimentación del sistema de control, que enlaza la función de transferencia lineal del proceso con el controlador borroso no lineal, da lugar a un lazo de control no lineal estándar que puede ser analizado mediante el criterio de Estabilidad Absoluta con un paradigma de dominio finito, por ejemplo, con el criterio del círculo [252].

#### 5.4 Control de un proceso de fabricación mediante sistema de inferencia borrosa optimizado a través del método de temple simulado

El sistema de control borroso optimizado en la sección anterior, también ha sido objeto de ajuste mediante el algoritmo de inspiración física temple simulado (SA, *Simulated Annealing*), que es un algoritmo de búsqueda meta-heurística para problemas de optimización global [17, 18]. El algoritmo minimiza la función de coste *ITAE*, mediante simulación del proceso de taladrado de alto rendimiento, controlado a lazo cerrado mediante el sistema de inferencia borroso de 49 reglas. La optimización del sistema de control (5.16) se logra ajustando las ganancias de entrada del controlador borroso  $[K_e \ K_{ce}]$ . El sistema de control es simulado, para comprobar su desempeño, empleando un modelo del proceso de TAR y seguidamente se realiza su implementación en tiempo real mediante ensayos experimentales.

$$K_{FCS} = [K_e, K_{ce}]_{OPT} = \arg \min (ITAE) \quad (5.16)$$

El algoritmo de SA comienza su ejecución con unos parámetros del controlador borroso iniciales  $\mathbf{K}_{FCS} = [K_e \ K_{ce}]_i$ , se simula el sistema de control con estos parámetros y seguidamente se evalúa la función de coste *ITAE* a partir de la señal de error. A continuación los parámetros del controlador  $\mathbf{K}_{FCS}$  iniciales son perturbados, generándose otro conjunto de parámetros  $\mathbf{K}_{FCS_{NEW}}$ , simulando nuevamente el sistema de control y evaluando el criterio *ITAE*. El criterio de aceptación o de rechazo se basa en el algoritmo de Metropolis [222]. El algoritmo de SA simula el proceso de templado de metales en su búsqueda de la solución óptima [253]. La perturbación aleatoria generada en los parámetros  $[K_e \ K_{ce}]$  provoca por ende un cambio en el sistema de control que se ve reflejado en el criterio *ITAE*. Las perturbaciones que se introducen dependen de un índice de temperatura  $T$  y de una constante de reducción de temperatura o velocidad de reducción  $\alpha$  (planificación de la temperatura de forma exponencial).

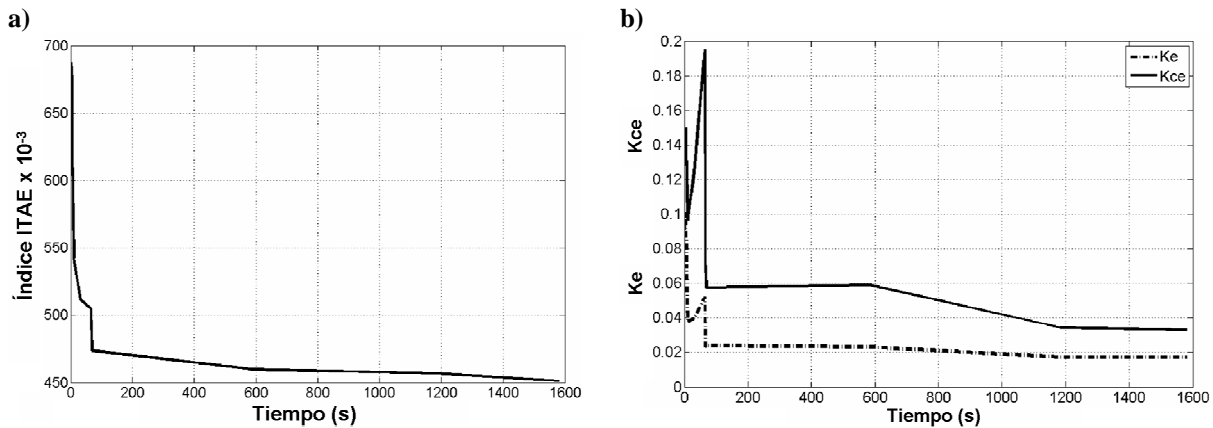
### 5.4.1 Optimización y simulación del controlador borroso

Como primer paso, el controlador borroso de 49 reglas se verifica mediante simulaciones en Matlab/Simulink, empleando el modelo dinámico de tercer orden del proceso de taladrado de alto rendimiento, presentado en el Capítulo 2. Los parámetros del controlador se han ajustado a un valor inicial de  $[K_e \ K_{ce} \ GC] = [0.5, 0.5, 1]$ , dando como resultado un índice de desempeño *ITAE* de 0.6353 y un máximo sobrepaso de 5.24%.

Para la aplicación del algoritmo, sus parámetros son ajustados previamente. Se fija la constante de reducción  $\alpha = 0.96$  y la temperatura inicial  $T_i = 200$ . Adicionalmente, se establece el número de perturbaciones aleatorias propuestas por cada temperatura en 100 y el número de soluciones aceptadas a las perturbaciones aleatorias propuestas en 30.

Al optimizar el sistema de control, pero solo ajustando las ganancias de entrada del controlador y manteniendo constante su ganancia de salida, es decir  $GC = 1$ , se obtiene  $[K_e \ K_{ce}]_{OPT} = [0.0173, 0.0333]$ , con un índice *ITAE* mínimo igual a 0.4508 y un correspondiente máximo sobrepaso de la respuesta transitoria del sistema de control igual a 0.65%.

La Figura 5.24 muestra el comportamiento del índice de desempeño *ITAE* y la variación de los parámetros del controlador, en cada una de las iteraciones de la ejecución del algoritmo de temple simulado.



**Figura 5.24. Optimización mediante temple simulado.**  
a) Índice *ITAE* y b) variación de los parámetros del controlador borroso.

El algoritmo fue ejecutado 75 veces consecutivas con el objetivo de minimizar la función de coste *ITAE*. Los resultados se muestran en la Figura 5.25. El comportamiento del índice de desempeño *ITAE* se representa en la Figura 5.25.a). Al término de la optimización, el valor máximo *ITAE* se alcanzó en la ejecución número 28 del algoritmo, siendo 3.77% mayor que

el valor mínimo alcanzado, sin embargo el error es menor del 2% en el resto de las ejecuciones. La Figura 5.25. b) muestra los valores correspondientes de los parámetros del controlador durante cada ejecución del algoritmo. Por otra parte, el tiempo de medio de cómputo del algoritmo de temple simulado es de 925 s, el cual se corresponde a su ejecución en la herramienta computacional Matlab, versión 7.1, en un ordenador con un procesador Pentium IV a 3.2 GHz de velocidad y 1.99GB de memoria RAM.

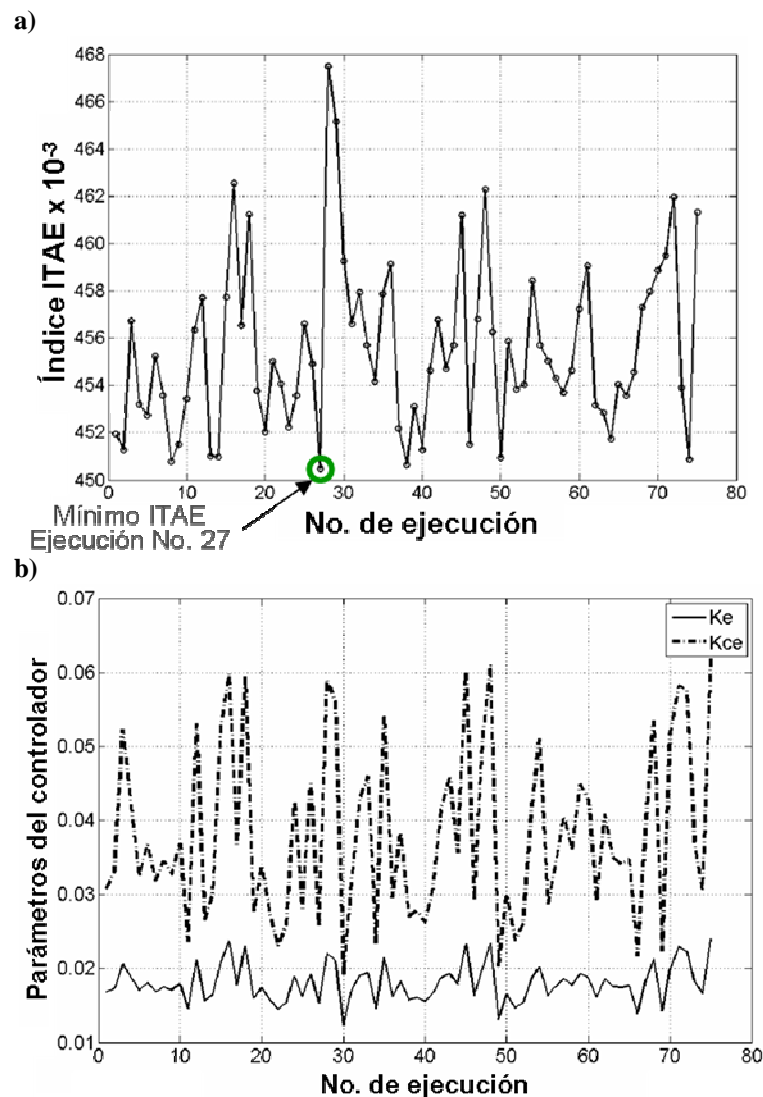


Figura 5.25. Resultados tras 75 ejecuciones del algoritmo de SA.

a) Variaciones del índice *ITAE* y b) parámetros del controlador [ $K_e$   $K_{ce}$ ].

Los resultados de la simulación del sistema de control borroso se muestran en la Figura 5.26, ilustrándose el comportamiento de la fuerza corte como variable controlada y la velocidad de avance de referencia como acción de control. Se ha simulado el sistema con sus parámetros iniciales y los obtenidos tras la optimización mediante temple simulado. Se puede

observar claramente en la figura que la respuesta transitoria, el tiempo de establecimiento y el máximo sobrepaso del sistema de control, son mejores después de la optimización.

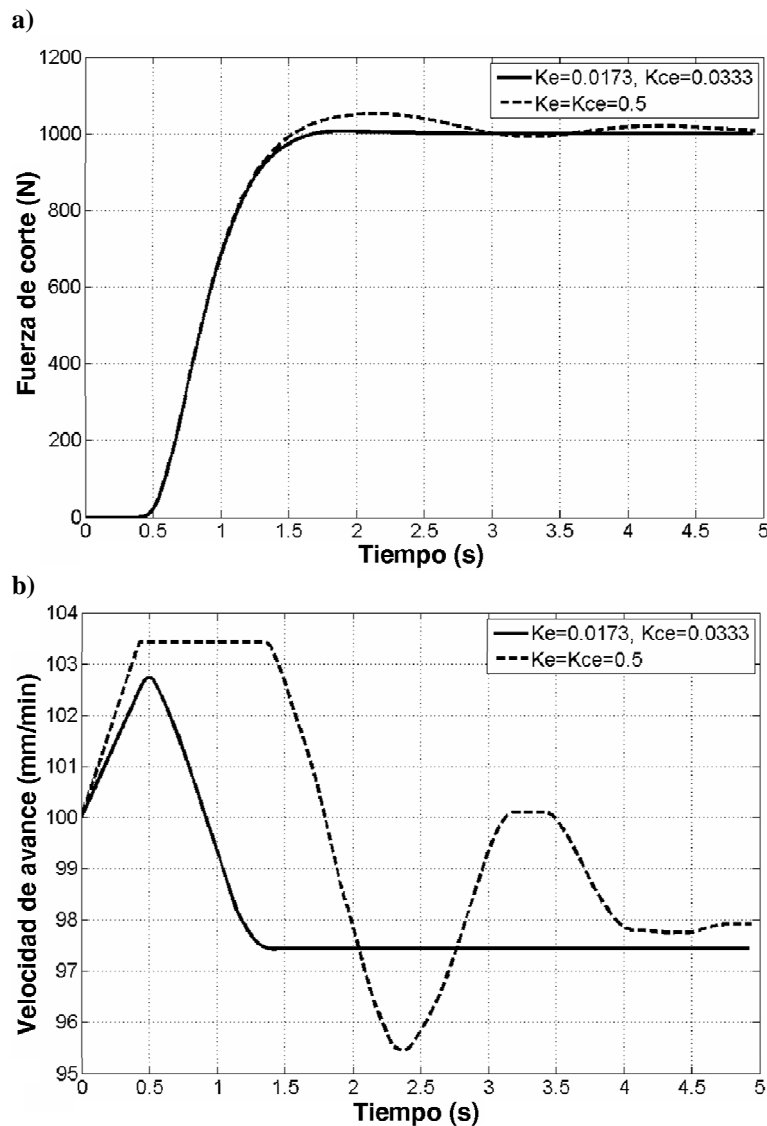


Figura 5.26. Simulación del sistema de control borroso optimizado mediante SA.  
a) Comportamiento de la fuerza de corte y b) velocidad de avance de referencia.

Con la finalidad de evaluar la influencia de la variación del tiempo de retardo del proceso en el desempeño de controlador, se ha llevado a cabo un estudio mediante simulación. Para simplificar, se ha asumido durante la simulación que el tiempo de retardo del sistema de control distribuido  $L$ , toma valores uniformemente distribuidos entre 0 y 0.8 s, ejecutándose 100 pruebas de simulación empleando el sistema de control borroso controlador optimizado. La Figura 5.27 describe el comportamiento de la fuerza de corte para cada tiempo de retardo.



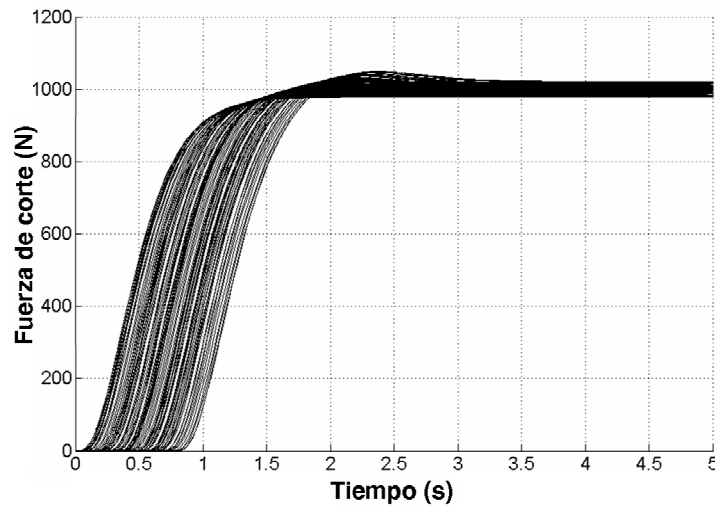


Figura 5.27. Comportamiento de la fuerza de corte según la variación del tiempo de retardo.

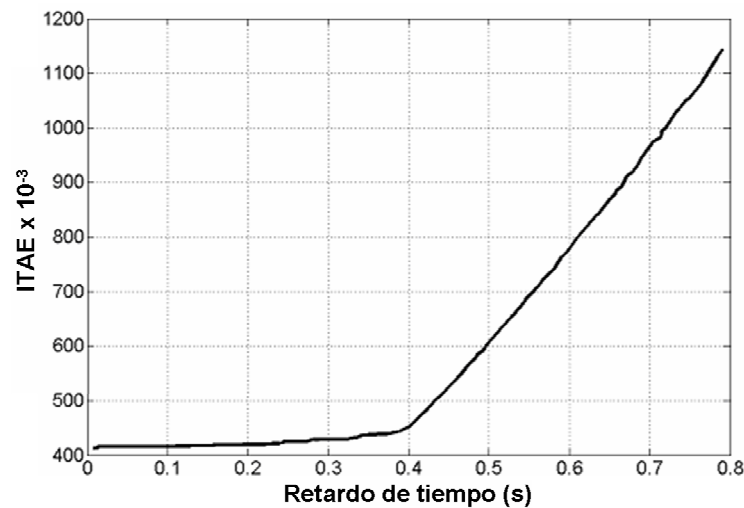


Figura 5.28. Comportamiento del criterio *ITAE* según varía el tiempo de retardo.

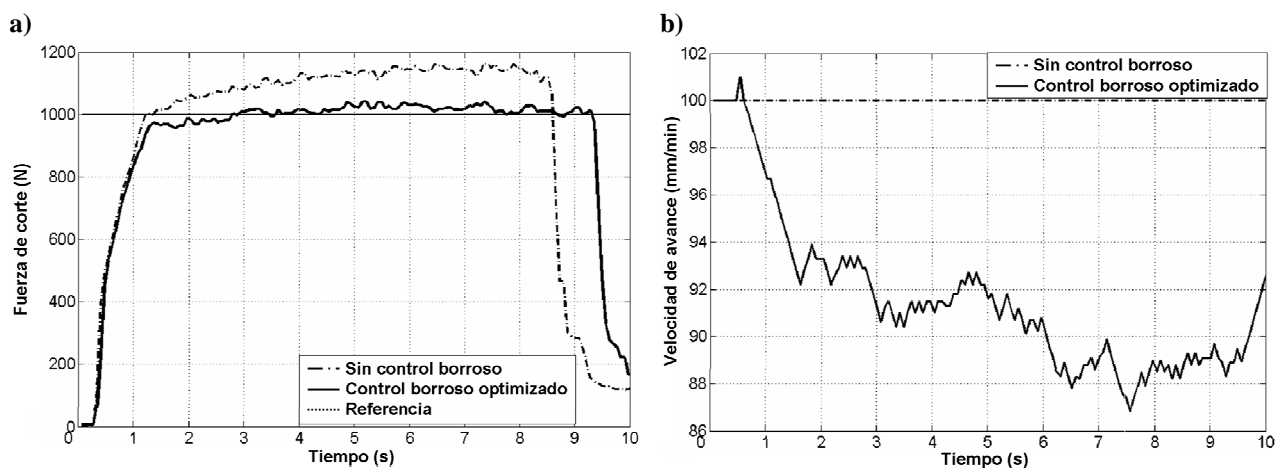
El comportamiento del índice de desempeño *ITAE*, según varía el retardo de tiempo del proceso, se muestra en la Figura 5.28. El valor mínimo del criterio *ITAE* se alcanza por el retardo mínimo y viceversa. Sin embargo, se consigue un buen comportamiento del índice *ITAE* aun suponiendo que el retardo máximo es mucho mayor que el utilizado para el modelo del proceso de taladrado. El peor de los casos entre todas las pruebas de simulación, se corresponde al máximo tiempo de retardo de 0.7954s, el cual provoca un máximo sobrepaso de 4.95% y un máximo *ITAE* igual a 1.1476.

### 5.4.2 Resultados experimentales

Empleando la plataforma experimental, se ha verificado de forma práctica el comportamiento del controlador borroso optimizado por entropía cruzada. Se han realizado taladros con una herramienta Sandvik de 10 mm de diámetro en piezas de hierro fundido nodular GGG40 con una dureza de 233 HB y de acero 17-4PH de una alta dureza igual a 31 HRC.

La fundición nodular GGG40 cada vez es más popular en varias aplicaciones ingenieriles debido a sus ventajas potenciales, como alta resistencia y tenacidad, baja fatiga y buena resistencia a desgaste. Por causa de las características metalúrgicas de este material, su mecanizado mediante técnicas convencionales como el fresado y el torneado, puede ser problemática y con un alto grado de dificultad. Por otra parte, el acero de alta dureza 17-4PH es utilizado en una gran variedad de aplicaciones entre las que se incluyen, por ejemplo, componentes de válvulas en la industria petrolera, equipamiento en procesos químicos, engranajes y muchas otras aplicaciones industriales. Las operaciones de taladrado, tanto controladas como no controladas, en materiales como el GGG40 y el acero 17-4 PH, constituye un campo incipiente de estudio para los investigadores.

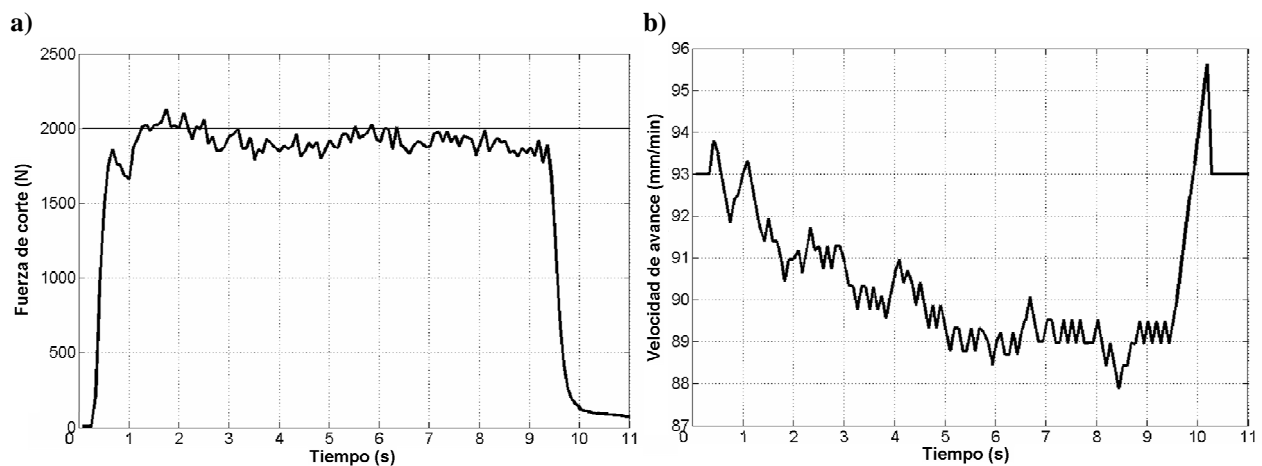
La velocidad de avance y la velocidad de giro nominales empleadas en los experimentos, recomendadas por el fabricante de la herramienta, han sido  $f_0=100\text{mm/min}$  y  $s_0=870\text{rpm}$ , respectivamente para el material GGG40 y  $f_0=93\text{mm/min}$  y  $s_0=780\text{rpm}$  para el 17-4PH. La profundidad máxima de la operación de taladrado es de 14 mm y el valor de referencia de la fuerza de corte es de 1000 N para el GGG40 y 2000 N para el 17-4PH.



**Figura 5.29.** Respuesta temporal del sistema de control optimizado mediante SA durante el taladrado del material GGG40.

Figura 5.29 y Figura 5.30 muestran los resultados experimentales de la operación de taladrado con la herramienta de 10 mm en piezas de los materiales GGG40 y 17-4PH. En la

Figura 5.29 se representa el comportamiento de la fuerza controlada y el cambio en la velocidad de avance durante el taladrado del material GGG40. De la misma forma, la Figura 5.30 muestra la respuesta temporal de la operación de taladrado en el material 17-4PH, empleando el controlador borroso optimizado mediante el algoritmo de temple simulado. Con el objetivo de compensar el incremento de la fuerza de corte a medida que se incrementa la profundidad de corte, como se puede observar en las gráficas, la velocidad de avance decrece gradualmente, de esta forma la fuerza de corte se mantiene regulada alrededor de su valor de referencia.



**Figura 5.30. Comportamiento del sistema de control optimizado durante una operación de taladrado del material 17-4PH**

El buen comportamiento de la respuesta transitoria del sistema de control es cuantificable a través de los índices de comportamiento. Durante el taladrado del material GGG40 se produce un máximo sobrepaso de un 4.18% y un índice *ITAE* igual a 35.81; en cambio, al realizar la operación de taladrado controlada por la estrategia borrosa optimizada en el material 17-4PH, la fuerza de corte produce un máximo sobrepaso igual al 6.44 % y el criterio de desempeño *ITAE* es igual a 82.55. La idoneidad del algoritmo de temple simulado para ajustar de forma óptima los sistemas de control borrosos, queda corroborada mediante los resultados experimentales mostrados.

## 5.5 Optimización de un sistema de inferencia borrosa mediante algoritmos genéticos para el control de un proceso de fabricación

El controlador borroso ajustado en la sección anterior, también ha sido objeto de optimización mediante otras estrategias inspiradas en la evolución biológica, como es el caso de los algoritmos genéticos (GA) [36]. De la misma forma, el algoritmo minimiza la función de coste *ITAE*, en base a simulación del proceso de taladrado de alto rendimiento, controlado a lazo cerrado mediante un sistema de inferencia borroso [254]. Figura 5.31 muestra el

diagrama de bloques del sistema de control borroso y el método de optimización de algoritmos genéticos, el cual utiliza como parámetros de entrada a la función de coste, las ganancias del controlador  $[K_e \ K_{ce}]$ .

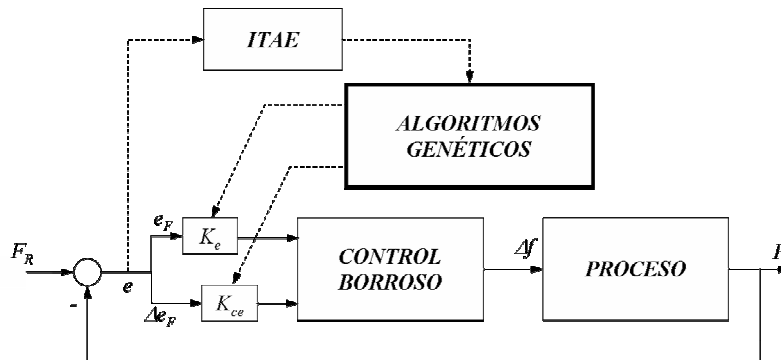
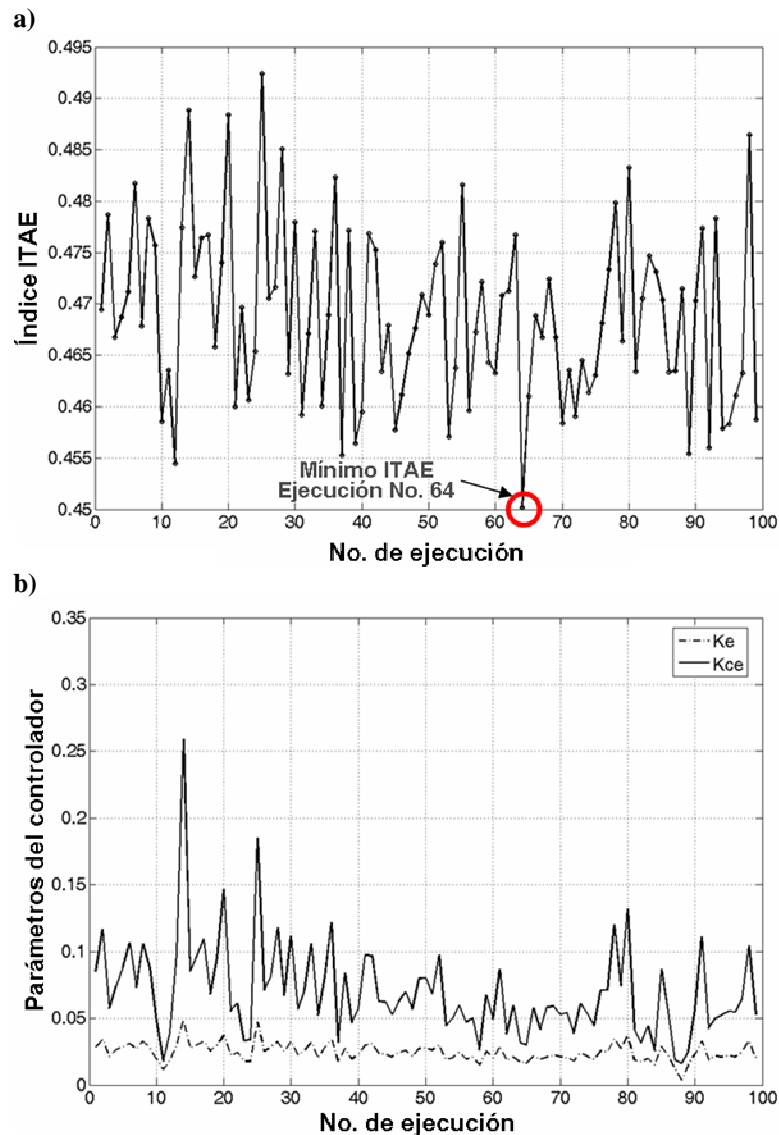


Figura 5.31. Diagrama del sistema de control borroso y optimización mediante algoritmos genéticos.

Antes de ejecutar el algoritmo de optimización, se ajustaron previamente los parámetros de cada una de sus etapas de selección, reproducción, mutación, cruzamiento y migración. La población inicial fue ajustada a 250 individuos, empleándose una función estocástica uniforme durante la etapa de selección para escoger los padres de la siguiente generación. En la etapa de reproducción solamente 5 individuos sobrevivirán a la próxima generación, con una tasa de cruzamiento del 60%. Por otra parte, para el proceso de mutación fue empleada una función de distribución Gaussiana y el cruzamiento fue realizado mediante diseminación. Durante la etapa de migración, el 40% de individuos en cada sub-población migran hacia una sub-población diferente, lo cual ocurre cada 10 generaciones. Por último, el algoritmo se detiene en un máximo de 500 iteraciones, pero si no hay mejora en el valor mínimo de la función de coste *ITAE*, el algoritmo culminará su ejecución en 150 iteraciones o hasta un tiempo de ejecución máximo de 50 segundos.

El algoritmo fue ejecutado 99 veces consecutivas para reducir al mínimo la función de coste. Los resultados se muestran en la Figura 5.32. El comportamiento del índice de desempeño *ITAE* se representa en la Figura 5.32.a. Al término de la optimización, el valor mínimo  $ITAE = 0.4513$  se alcanzó en la ejecución número 64 del algoritmo, correspondiéndose a unos valores de los parámetros del controlador borroso  $[K_e \ K_{ce}]_{OPT} = [0.0169, 0.0314]$ , produciendo un máximo sobrepaso de 0.57%. La Figura 5.32.b muestra los valores correspondientes de los parámetros del controlador durante cada ejecución del algoritmo. Por otra parte, el tiempo de cómputo total del algoritmo fue de 238.6 s, el cual se corresponde a su ejecución empleando las funciones del paquete de algoritmos genéticos de Matlab en su versión 7.1, en un ordenador con un procesador Pentium IV a 3.2 GHz de velocidad y 1.99GB de memoria RAM.



**Figura 5.32. Resultados tras 99 ejecuciones de la optimización mediante GA**  
a) Variaciones del índice *ITAE* y b) variaciones de los parámetros  $[K_e \ K_{ce}]$ .

El sistema de control fue simulado en Matlab/Simulink para las ganancias del controlador ajustadas. Los resultados de la simulación son mostrados en la Figura 5.33, en ella se puede observar el comportamiento de la fuerza de corte ante un cambio de tipo de escalón en el valor de referencia y la velocidad de avance, como señal de control, la cual se expresa en valor porcentual, referido a su valor nominal  $f_0 = 100 \text{ mm/min}$ . Durante la simulación se obtuvo un máximo sobrepaso igual a 0.5659%.

Con el fin de analizar las capacidades de rechazo ante perturbaciones del sistema de control optimizado, se ha considerado la presencia de ruido aditivo durante la simulación (ver diagrama en la Figura 5.34), el cual puede representar la influencia de dinámicas no

modeladas en el proceso de TAR en tiempo real. Se asume la siguiente función como perturbación aditiva en la salida del proceso:

$$d(t) = A \cdot (\sin \omega_1 t + \sin \omega_2 t + \sin \omega_3 t + \sin \omega_4 t) \quad (5.17)$$

donde la amplitud de la perturbación aditiva es  $A=0.2$ , que corresponde a un 20% de variación de la ganancia del modelo del proceso y las frecuencias  $[\omega_1 \ \omega_2 \ \omega_3 \ \omega_4] = [3.98 \ 12.94 \ 22.89 \ 27.87]$  son armónicos superiores de las frecuencias de los polos del modelo.

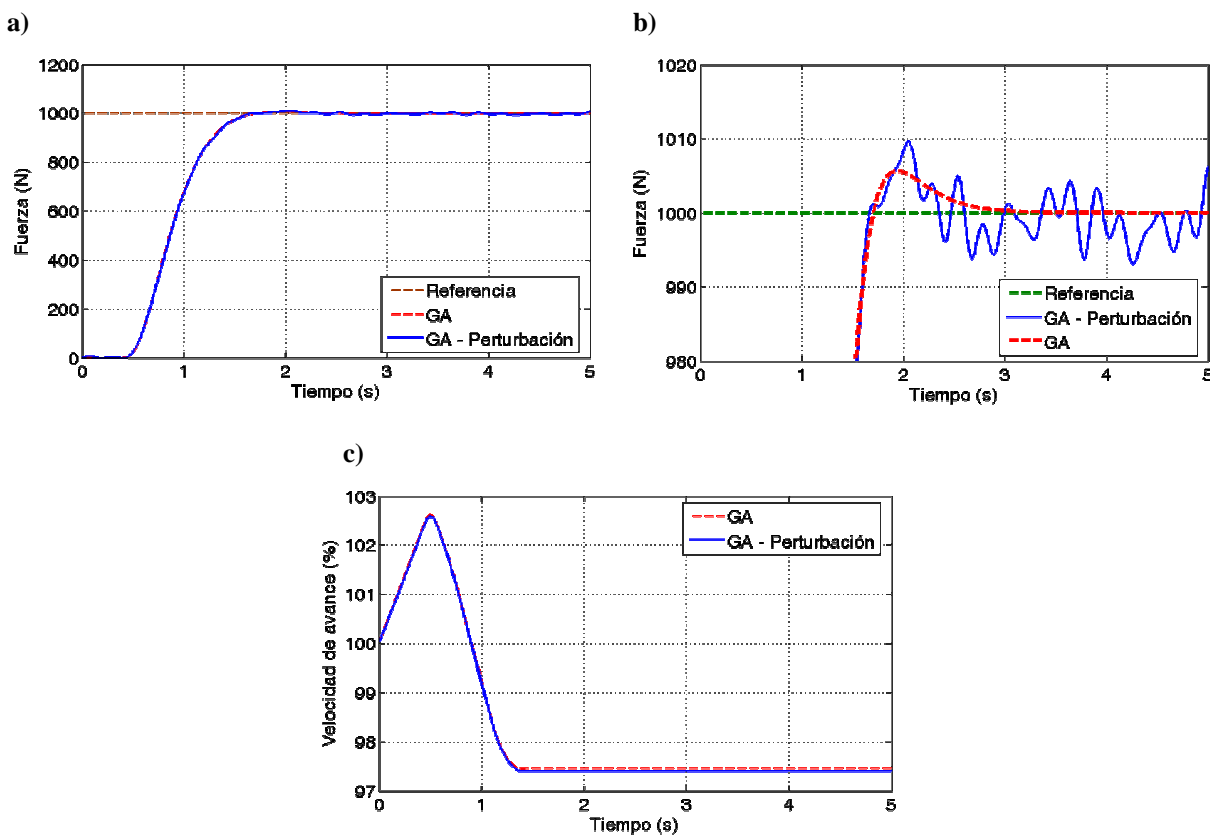


Figura 5.33. Simulación del sistema de control optimizado mediante GA.

a) Fuerza de corte, b) fuerza de corte ampliada entre 980 y 1020 N y c) velocidad de avance de referencia.

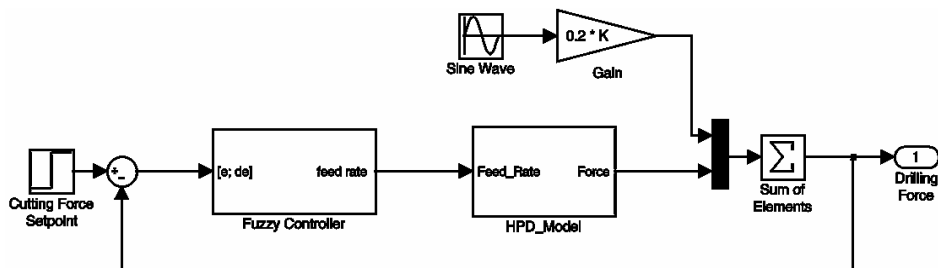


Figura 5.34. Aplicación de simulación del sistema de control con ruido aditivo.

En la Figura 5.33 se ilustra también el comportamiento del sistema de control en presencia de perturbaciones. Como puede observarse con un mayor detalle en la Figura 5.33.b la respuesta temporal cambia ligeramente, obteniéndose para estas condiciones un índice *ITAE* igual a 0.4763 y un máximo sobrepaso de un 0.9633% pero, como se aprecia, el sistema de control es capaz de regular el proceso, de forma satisfactoria, aún en presencia de ruido.

## 5.6 Ajuste de un sistema de inferencia borrosa, mediante el método de entropía cruzada, para el control de un proceso de fabricación

En esta sección se propone una estrategia simple y flexible, que emplea un método de optimización basado en población y simulación, para ajustar un controlador borroso. Se aplica el algoritmo de entropía cruzada (CE) para ajustar los factores de escala de entrada del controlador borroso de 49 reglas diseñado con anterioridad, a partir de la minimización de una función de coste o índice de desempeño [254]. El sistema de control tiene como objetivo el control de la fuerza de corte en el proceso de taladrado de alto rendimiento.

Un criterio de gran utilidad que penaliza transitorios de larga duración en la respuesta temporal de un sistema de control, es la integral del valor absoluto del error ponderado en el tiempo (*ITAE*):

$$ITAE = \int_0^T |e(t)| \cdot t dt \quad (5.18)$$

Esta función de coste o índice de desempeño será considerado como el problema de optimización del método CE (4.37). El criterio *ITAE* es más selectivo que otros, como la integral del error absoluto (*IAE*) y la integral del error cuadrático (*ISE*), produciendo, además, menor sobrepaso y oscilaciones que estos criterios.

Otro criterio que se tendrá en cuenta durante la optimización del sistema de control borroso es la integral del error cuadrático ponderado por el tiempo al cuadrado (*ISTSE*). En un sistema de control a lazo cerrado, grandes cambios en la carga como perturbación, producen grandes valores de error, por tanto, es de gran utilidad disponer de un criterio que penalice ligeramente los errores iniciales (instantes de tiempo iniciales) y que penalice duramente los errores en instantes de tiempos superiores, durante la respuesta transitoria del sistema de control. En muchos casos el criterio *ISTSE* constituye además, una elección natural para optimizar el sistema de control sin oscilaciones y sobrepaso en su respuesta temporal.

$$ISTSE = \int_0^T e^2(t) \cdot t^2 dt \quad (5.19)$$

El criterio *ITAE* se utiliza en el método de entropía cruzada como función de coste, en detrimento del resto de los índices de desempeño mencionados con anterioridad, por la buena selectividad que proporciona. Para el método se emplea como función de distribución probabilística la distribución normal, con un ajuste dinámico de sus parámetros y siendo los factores de escala las medias de las distribuciones a estimar. El algoritmo de CE comienza con unos parámetros del controlador borroso iniciales  $\mathbf{K}_{FCS} = [K_e \ K_{ce}]_i$  y evalúa el índice *ITAE* (5.18). Entonces el problema de optimización se traduce en:

$$K_{FCS} = [K_e, K_{ce}]_{OPT} = \arg \min (ITAE) \quad (5.20)$$

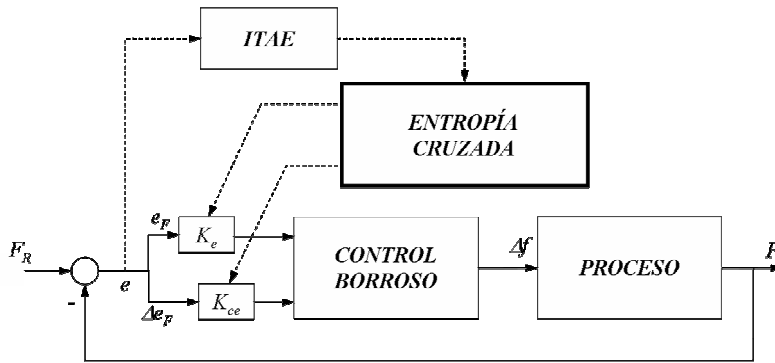


Figura 5.35. Diagrama del sistema de control borroso y el método de optimización de entropía cruzada.

La Figura 5.35 muestra el diagrama de bloques del sistema de control borroso. El ajuste óptimo de los parámetros del controlador se realiza mediante la minimización del criterio *ITAE*. Tras ejecutar el método de entropía cruzada se obtiene como resultado los parámetros del controlador óptimos. En el diagrama de bloques, las líneas discontinuas representan las “señales” más representativas utilizadas para la optimización.

### 5.6.1 Resultados de la simulación

El controlador borroso introducido en secciones anteriores se ha verificado mediante simulaciones en Matlab/Simulink, empleando el modelo dinámico de tercer orden del proceso de taladrado de alto rendimiento, presentado en el Capítulo 2, sección 2.3. Los parámetros del controlador se han ajustado a un valor inicial de  $[K_e \ K_{ce}]_{INI} = [0.5, 0.5]$ . Después de la simulación del sistema de control, empleando estos parámetros, se obtiene un índice de desempeño *ITAE* de 0.64 y un máximo sobrepaso de 5.24%.

El procedimiento de entropía cruzada se representa en la Figura 5.36, empleando como valores iniciales para la media y desviación estándar de las funciones probabilísticas,  $\hat{\mu}_0 = 0$  y  $\hat{\sigma}_0 = 0.5$ , además como parámetros del algoritmo para el ajuste dinámico  $\alpha = 0.5$ ,  $\beta = 0.92$



y  $q = 2$ , un número de muestras elites  $N^{elite} = 4$  y un total de muestras aleatorias en cada iteración  $N = 100$  (conjunto de parámetros del controlador  $[K_e^i, K_{ce}^i]_{i=1 \dots N}$ ). El procedimiento de optimización finalizará cuando la desviación estándar sea menor a  $\varepsilon = 5 \cdot 10^{-4}$ , lo cual ocurre después de 77 iteraciones del algoritmo. En la Figura 5.36 se muestran las funciones de densidad probabilística para los parámetros  $K_e, K_{ce}$ , donde el mayor cúmulo de funciones se obtienen concentradas en las cercanías de  $[K_e, K_{ce}]_{OPT}$ .

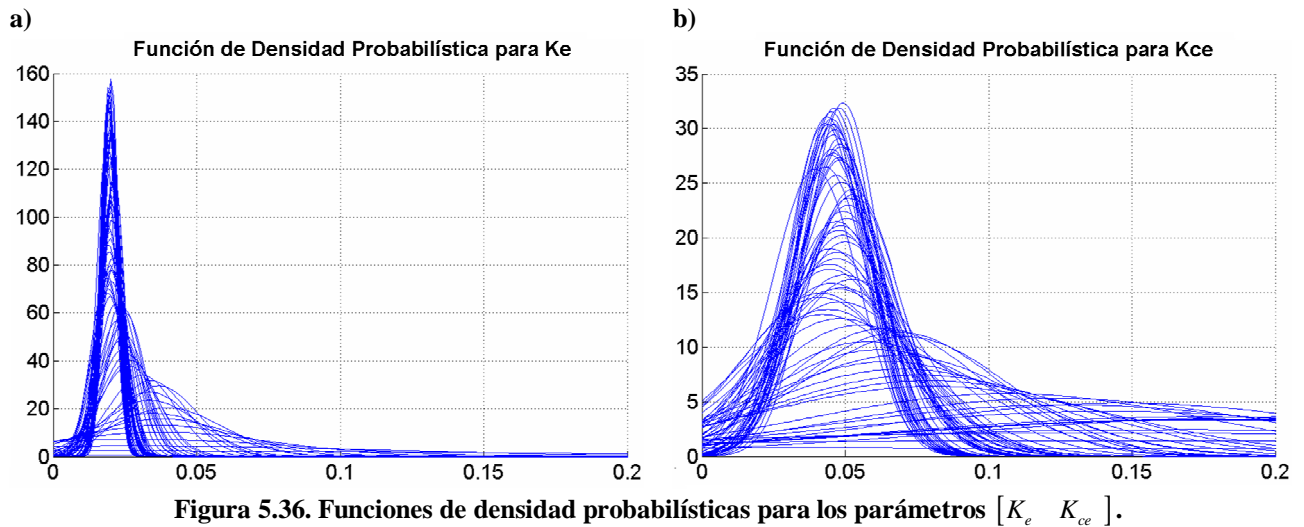
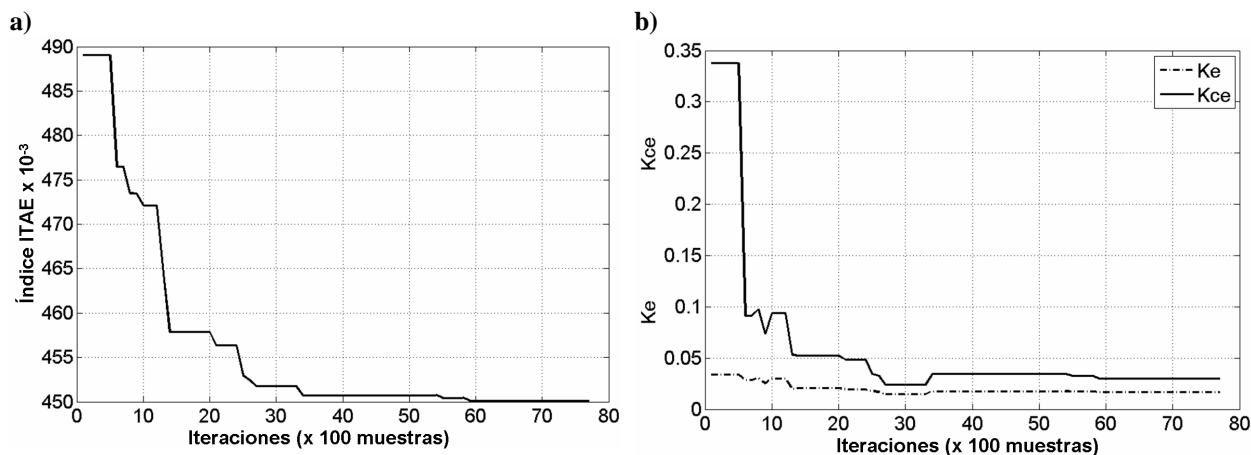


Figura 5.36. Funciones de densidad probabilísticas para los parámetros  $[K_e, K_{ce}]$ .

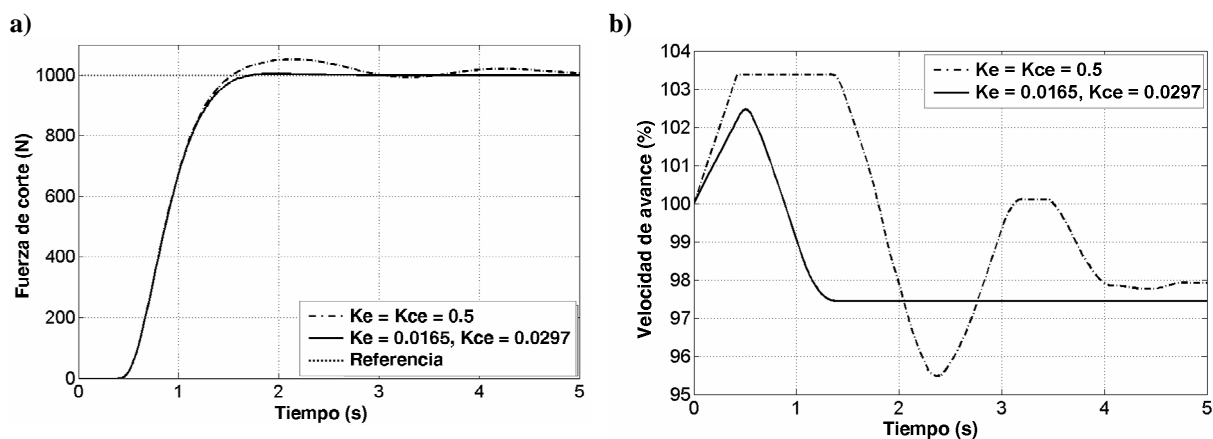
La optimización del sistema de control borroso basada en entropía cruzada devuelve como parámetros óptimos del controlador  $[K_e, K_{ce}]_{OPT} = [0.0165, 0.0297]$ , obteniéndose un índice de desempeño *ITAE* de 0.45 y un máximo sobrepaso de 0.46%. La Figura 5.37 muestra el comportamiento del índice de desempeño *ITAE* y la variación de los parámetros del controlador, en cada una de las iteraciones de la ejecución del algoritmo de entropía cruzada.

En la Figura 5.38 se muestran los resultados de la simulación del sistema de control borroso, caracterizado por el comportamiento de la fuerza corte como variable controlada y la velocidad de avance de referencia como acción de control. El sistema de control se ha simulado con sus parámetros ajustados con los valores iniciales  $[K_e, K_{ce}]_{INI}$  y los obtenidos tras la optimización con entropía cruzada,  $[K_e, K_{ce}]_{OPT}$ . Se puede observar claramente en la figura que la respuesta transitoria, el tiempo de establecimiento y el máximo sobrepaso, del sistema de control, son mejores después de la optimización con el algoritmo de entropía cruzada.



**Figura 5.37. Resultados durante la optimización mediante entropía cruzada.**

a) Índice de desempeño *ITAE* y b) parámetros del controlador borroso.



**Figura 5.38. Simulación del sistema de control borroso.**

a) Comportamiento de la fuerza de corte y b) velocidad de avance de referencia.

Con el objetivo de evaluar la influencia de la variación del tiempo de retardo del proceso en el desempeño del controlador, se ha llevado a cabo un estudio mediante simulación. Por motivo de simplicidad, se ha asumido durante la simulación, que el tiempo de retardo del sistema de control distribuido  $L$ , toma valores uniformemente distribuidos entre 0 y 0.8 s. Se han ejecutado 100 pruebas de simulación con los parámetros del controlador borroso ajustados a sus valores óptimos.

La Figura 5.39 describe el comportamiento de los índices de desempeños *ITAE* e *ISTSE*, según varía el retardo de tiempo del proceso. En el estudio se ha tenido en cuenta además el comportamiento del criterio *ISTSE*, para evaluar la influencia de los retardos en la sensibilidad de este índice de desempeño. El valor mínimo del criterio *ITAE* ha sido alcanzado por el retardo mínimo y viceversa. Sin embargo, los valores que alcanza el índice *ITAE* son buenos aun suponiendo que el retardo máximo es mucho mayor que el modelado experimentalmente (ver Capítulo 2, sección 2.3). La curva que describe el comportamiento

del criterio *ISTSE* muestra una pendiente muy cercana a cero hasta 0,4 segundos, lo cual simboliza que el criterio *ITAE* muestra mejor sensibilidad que el criterio *ISTSE* para este caso de estudio. Entre todas las pruebas de simulación, el peor caso se corresponde a un tiempo de retardo de 0.795s, donde se obtiene  $ITAE = 1.125$ ,  $ISTSE = 0.5695$  y un máximo sobrepaso de 4.454%.

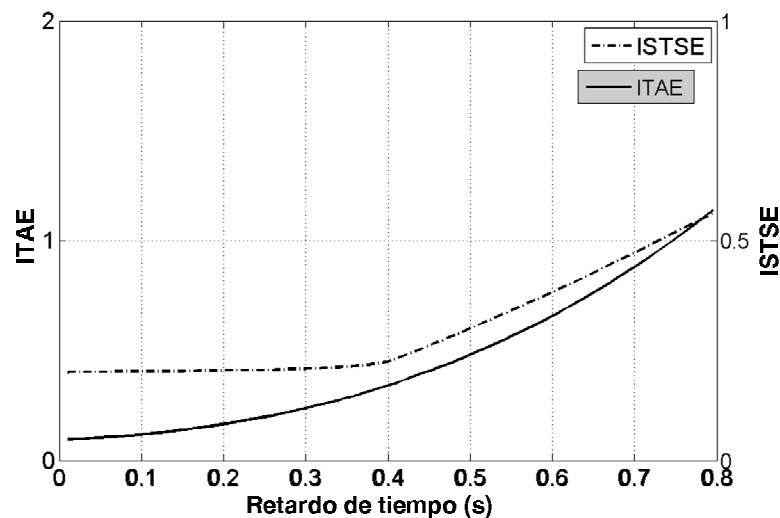


Figura 5.39. Comportamiento de los criterios de desempeño *ITAE* e *ISTSE* en presencia de retardos de tiempo en el sistema de control.

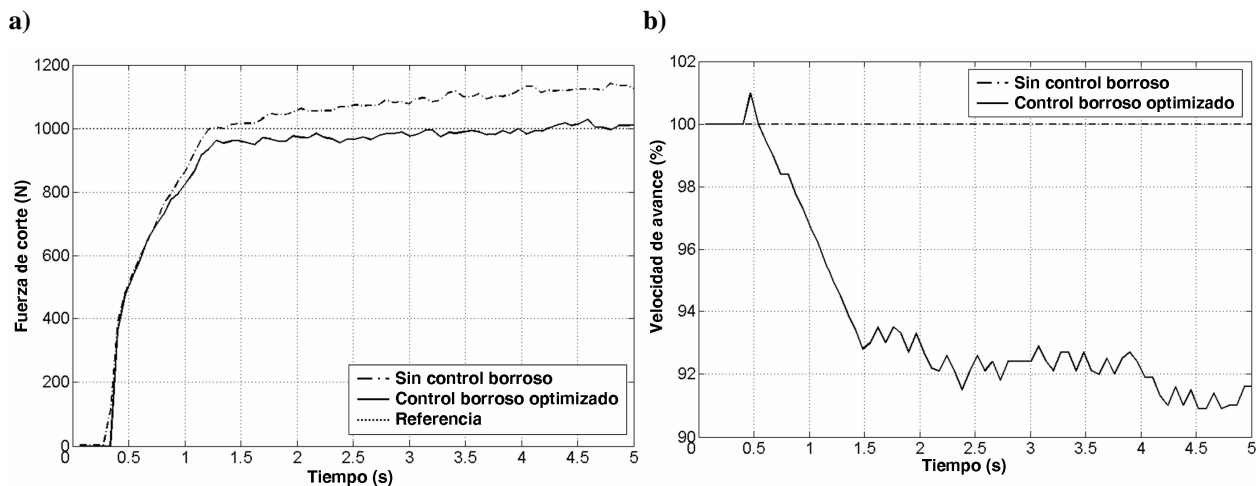
## 5.6.2 Resultados experimentales

Empleando la plataforma experimental, se ha verificado de forma práctica el comportamiento del controlador borroso optimizado por entropía cruzada. Se han realizado taladros con una herramienta Sandvik de 10 mm de diámetro en piezas de hierro fundido nodular GGG40 con una dureza de 233 HB.

La velocidad de avance y la velocidad de giro nominales empleadas en los experimentos, recomendadas por el fabricante de la herramienta, han sido  $f_0=100\text{mm/min}$  y  $s_0=870\text{rpm}$ , respectivamente. La profundidad máxima de la operación de taladrado es de 14 mm y el valor de referencia de la fuerza de corte es de 1000 N.

La Figura 5.40 muestra los resultados experimentales del taladrado con una herramienta de 10 mm en la pieza del material GGG40. La Figura 5.40a representa el comportamiento de la fuerza controlada y la Figura 5.40b el cambio en la velocidad de avance en valor porcentual referido al valor nominal, correspondiéndose ambas gráficas al sistema de control borroso optimizado por el método CE. Con el objetivo de compensar el incremento de la fuerza de corte a medida que se incrementa la profundidad de corte, como se puede observar en las gráficas, la velocidad de avance decrece gradualmente, de esta forma la fuerza de corte prácticamente no varía alrededor de su valor de referencia. El buen comportamiento de la

respuesta transitoria del sistema de control, está respaldado por los valores de los índices de desempeño, tal y como se muestra en la Tabla 5.7. Estos resultados experimentales corroboran la idoneidad del método de CE para el ajuste óptimo de los sistemas de control borrosos.



**Figura 5.40. Comportamiento del sistema de control durante el taladrado del material GGG40.**  
a) Fuerza de corte y b) velocidad de avance.

**Tabla 5.7. Estudio comparativo de los resultados experimentales y de simulación del comportamiento del controlador borroso optimizado mediante el método de entropía cruzada.**

Parámetros del controlador borroso	Procedimiento	ITAE	ITSE	ISTSE	Sobrepaso (%)
Sin compensación de fuerza	Experimental	1.376	0.243	0.552	14.28
$K_e = 0.5$ $K_{ce} = 0.5$	Simulación	0.636	0.306	0.185	5.67
$K_e = 0.0165$ $K_{ce} = 0.0297$	Simulación	0.450	0.302	0.171	0.462
	Experimental	0.449	0.131	0.673	2.841

## 5.7 Estudio comparativo entre las técnicas de optimización empleadas para el ajuste de estrategias borrosas de control de un proceso de fabricación complejo

El desempeño del sistema de control borroso de 49 reglas, ajustado mediante diferentes técnicas de optimización, será analizado mediante un estudio comparativo [254]. El estudio evaluará el desempeño de la estrategia de control optimizada mediante: el algoritmo de entropía cruzada (CE), el método bio-inspirado de algoritmos genéticos (GA), el algoritmo de temple simulado (SA) y el algoritmo de Nelder Mead (NM).

El estudio comparativo se ha realizado en base a simulaciones del comportamiento de las estrategias de control borroso, aplicadas al proceso de taladrado de alto rendimiento, el cual se representa por su modelo computacional desarrollado. La simulación se ha realizado empleando el programa Matlab, su versión 7.1 para Windows, en un ordenador con procesador Pentium IV de 3.2 GHz de velocidad y una memoria RAM de 1.99 GB.

**Tabla 5.8. Parámetros de las estrategias de optimización para un ajuste óptimo de un controlador borroso de 49 reglas.**

Método	Entropía cruzada	Temple Simulado	Nelder-Mead	Algoritmo genético
Parámetros de ajuste	1. $\alpha=0.5$ 2. $\beta=0.92$ 3. $q=2$ 4. $N^{elite}=4$ 5. $N=100$ 6. $\varepsilon=5 \cdot 10^{-4}$	1. $\alpha=0.96$ 2. $T_i=200$ 3. Perturb. aleat.: 100 4. Soluciones aceptadas: 30	1. MaxFunEvals = 400 2. MaxIter = 400 3. TolFun = 1.00e-004 4. TolX = 1.00e-004 5. $\begin{bmatrix} K_e \\ K_{ce} \end{bmatrix}_{INIT} = \begin{bmatrix} 1.0 \\ 1.0 \end{bmatrix}$	1. Población: 250 2. Función de selección: estocástica uniforme. 3. Sobrevivientes: 5 4. Tasa de cruzamiento: 60% 5. Función de mutación: Gaussiana 6. Función de cruzamiento: Diseminación 7. Intervalo de migración: 10 8. Tasa de migración: 40% 9. Generaciones: 500 10. Generaciones consecutivas sin mejora: 150 11. Tiempo sin mejora límite: 50s

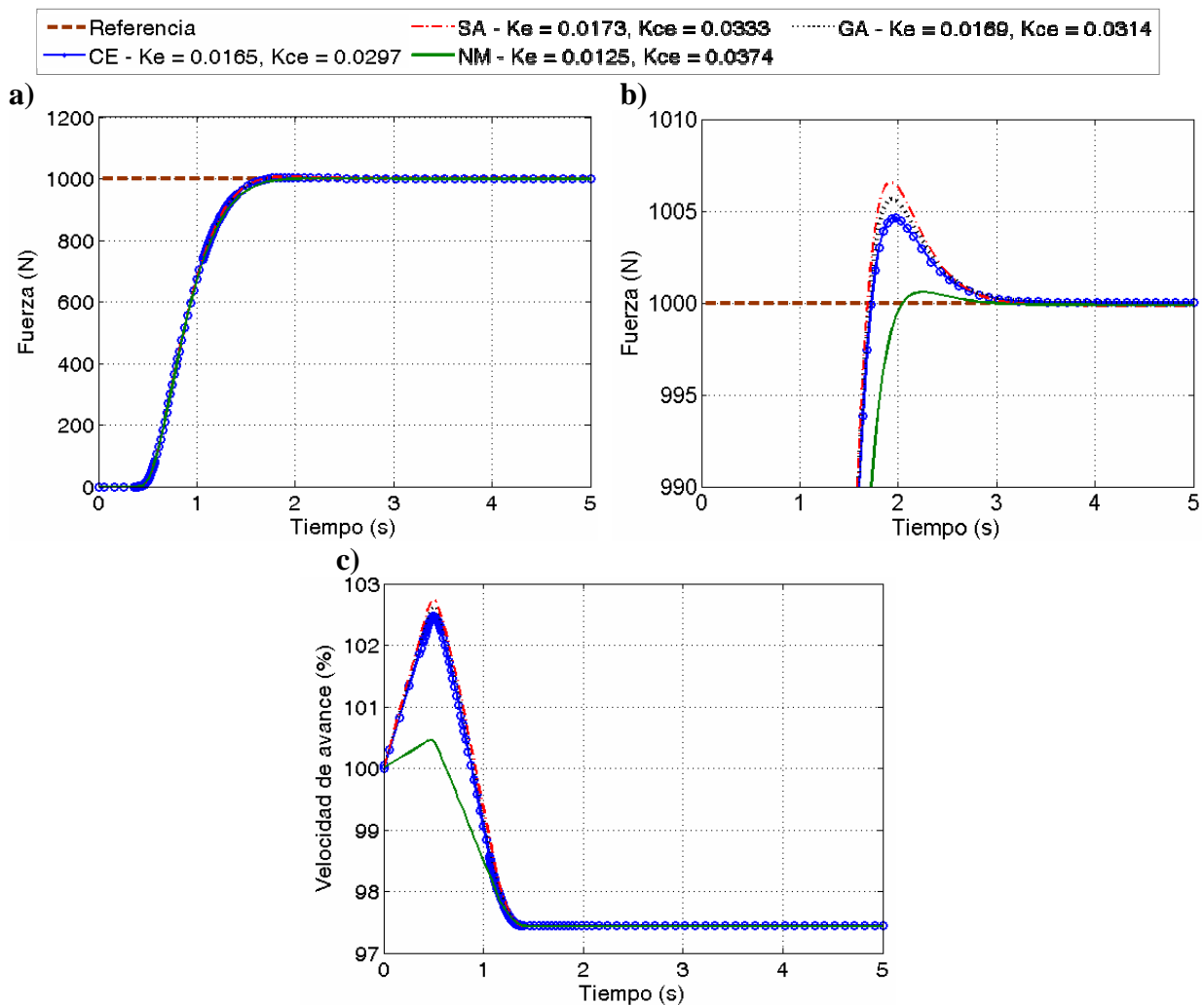
En la Tabla 5.8 se resumen las técnicas de optimización empleadas, así como los parámetros ajustados para cada una de ellas. En cada una de las estrategias se ha optimizado el controlador borroso de 49 reglas, minimizando como función de coste la integral del valor absoluto del error ponderado en el tiempo (*ITAE*) y utilizando los parámetros del controlador  $[K_e \ K_{ce}]$  como coeficientes de ajuste.

La Tabla 5.9 describe los parámetros de los controladores ajustados por los cuatro métodos de optimización, así como los índices de comportamiento *ITAE* y la integral del error cuadrático ponderado por el tiempo al cuadrado (*ISTSE*), además del máximo sobrepaso, que caracterizan la respuesta dinámica en el dominio temporal del sistema de control borroso. Con el objetivo de evaluar el coste computacional y el nivel de complejidad de cada método, la tabla también muestra el tiempo de cómputo y el número de parámetros de ajuste por cada método de optimización.

La respuesta temporal de los cuatro sistemas de control sin presencia de ruido añadido en la simulación es mostrada en la Figura 5.41. Las respuestas transitorias son bastante similares ya que, en los cuatro casos, se trata de parámetros óptimos o sub-óptimos de los sistemas de control borrosos.

**Tabla 5.9. Estudio comparativo del desempeño del sistema de control borroso sin presencia de perturbaciones.**

Método	Parámetros del controlador	ITAE	ISTSE	Sobre-paso (%)	Parámetros ajustados	Tiempo Cómput. (s)
Entropía Cruzada (CE)	$K_e = 0.0165$ $K_{ce} = 0.0297$	0.4502	0.1716	0.4675	6	369.50
Temple Simulado (SA)	$K_e = 0.0173$ $K_{ce} = 0.0333$	0.4508	0.1712	0.6705	4	925.00
Nelder-Mead (NM)	$K_e = 0.0125$ $K_{ce} = 0.0374$	0.4604	0.1748	0.0732	5	156.03
Algoritmos Genéticos (GA)	$K_e = 0.0169$ $K_{ce} = 0.0314$	0.4513	0.1714	0.5659	11	238.60

**Figura 5.41. Estudio comparativo sin presencia de perturbaciones.**

a) Fuerza de corte, b) fuerza ampliada alrededor del valor de referencia y c) señal de control velocidad de avance.

Con el fin de analizar las capacidades de rechazo ante perturbaciones de los sistemas de control, se ha considerado durante la simulación, la presencia de ruido aditivo, además de la influencia de dinámicas no modeladas, de tal manera que el comportamiento dinámico del sistema se aproxime al comportamiento en tiempo real del proceso controlado. Se asume la siguiente función como perturbación aditiva en la salida del proceso:

$$d(t) = A \cdot (\sin \omega_1 t + \sin \omega_2 t + \sin \omega_3 t + \sin \omega_4 t) \quad (5.21)$$

donde la amplitud de la perturbación aditiva es  $A=0.2$ , que corresponde a un 20% de variación de la ganancia del modelo del proceso y las frecuencias  $[\omega_1 \ \omega_2 \ \omega_3 \ \omega_4] = [3.98 \ 12.94 \ 22.89 \ 27.87]$  son armónicos superiores de los polos del modelo.

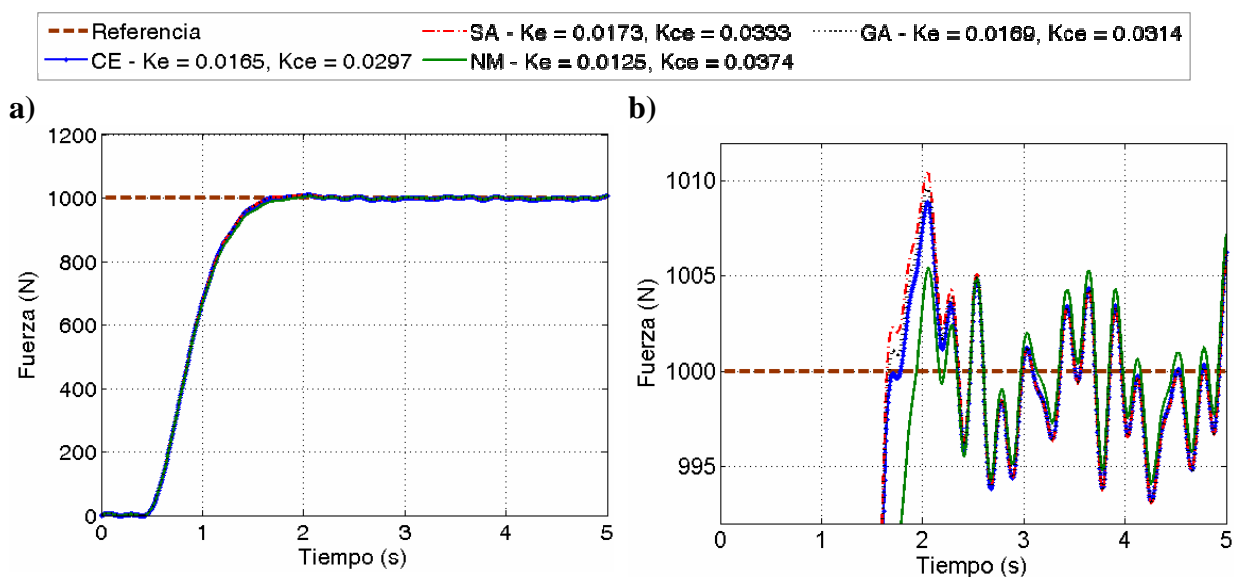
El comportamiento de los sistemas de control cambia ligeramente ante presencia de perturbaciones, pero aún son capaces de regular el proceso en presencia de ruido tal y como puede observarse en la Tabla 5.10 y en la Figura 5.42. Según el criterio *ITAE*, los parámetros del controlador optimizados mediante el método de entropía cruzada, son los que ofrecen la mejor respuesta transitoria y el segundo mejor sobrepaso después del método de Nelder-Mead. Por otra parte, el criterio *ISTSE* no varía significativamente entre todos los sistemas de control.

**Tabla 5.10. Estudio comparativo del desempeño del sistema de control borroso en presencia de perturbaciones.**

Estrategia de optimización	Parámetros del controlador	<i>ITAE</i>	<i>ISTSE</i>	Sobrep. (%)
Entropía Cruzada (CE)	$K_e = 0.0165$ $K_{ce} = 0.0297$	0.4760	0.1722	0.8873
Temple Simulado (SA)	$K_e = 0.0173$ $K_{ce} = 0.0333$	0.4767	0.1718	1.0443
Nelder-Mead (NM)	$K_e = 0.0125$ $K_{ce} = 0.0374$	0.4855	0.1753	0.5427
Algoritmos Genéticos (GA)	$K_e = 0.0169$ $K_{ce} = 0.0314$	0.4763	0.1720	0.9633

A pesar de que el ruido introducido, empeora el desempeño de los sistemas de control, según el análisis, el método de entropía cruzada ofrece el mejor *ITAE* entre todos los métodos,

con una buena relación entre el tiempo de cómputo y los parámetros de ajuste, el algoritmo de entropía cruzada es más rápido, en cuanto tiempo de cómputo se refiere, que el método de temple simulado y menor que algoritmos genéticos y, por otra parte, tiene menos parámetros de ajuste que el método de algoritmos genéticos. Sin embargo, se ha de destacar que el tiempo de cómputo es un criterio un tanto engañoso, porque no se ha tenido en cuenta para la comparación, por ejemplo, el tiempo total empleado para obtener los parámetros de ajuste más adecuados.



**Figura 5.42. Estudio comparativo en presencia de perturbaciones.**  
a) Fuerza de corte y b) fuerza ampliada alrededor del valor de referencia.

## 5.8 Conclusiones

En este capítulo se han diseñado sistemas de control optimizados, basados en técnicas clásicas y de *soft-computing*, para su aplicación en la regulación automática de procesos complejos, comprobándose su desempeño en una plataforma experimental compuesta por un centro de mecanizado de alta velocidad, en específico durante el control de operaciones de taladrado de alto rendimiento.

El diseño de las estrategias de control se ha basado en una metodología en la que, a partir del empleo de modelos computacionales de los procesos físicos, inicialmente se realiza su ajuste mediante simulación y seguidamente se efectúa su implementación y evaluación en tiempo real directamente en el proceso, a través de una plataforma experimental.

Se han diseñado estrategias de control clásicas lineales, como el algoritmo de control proporcional, integral y derivativo, PID, de muy amplia utilización en el control de procesos



físicos en general, y basadas en *soft-computing*, como es el caso de los sistemas de inferencia borrosa y los sistemas híbridos neuro-borrosos.

El ajuste del controlador PID se ha realizado utilizando un método clásico como Ziegler-Nichols, el cual se basa en el conocimiento del comportamiento dinámico del proceso a controlar, haciendo necesario el empleo de un modelo lineal.

Además, en el caso de las estrategias de control basadas en técnicas de *soft-computing*, inicialmente se han utilizado sistemas de inferencia borrosa híbridos de tipo ANFIS, como modelos directo e inverso de una estructura de control por modelo interno. Para corroborar el buen comportamiento del sistema de control, se ha comparado su desempeño con la estrategia de control lineal y con el proceso a lazo abierto, es decir, sin realimentación de la variable controlada.

Por otra parte, se ha diseñado un sistema de control borroso que consta de dos entradas, una salida y 49 reglas en su sistema de inferencia. El controlador se ha ajustado variando los factores de escala de sus entradas, utilizando para ello algoritmos de optimización de inspiración física y biológica, como ha sido el caso del algoritmo heurístico de Nelder-Mead, el método de inspiración física temple simulado, de inspiración biológica como son los algoritmos genéticos y el método de entropía cruzada. En todos los casos el ajuste se ha realizado minimizando como función objetivo la integral del error absoluto ponderado en el tiempo (*ITAE*), debido a que penaliza las respuestas transitorias temporales de larga duración y es muy sensible ante variaciones de parámetros en comparación con otros criterios, como *IAE*, *ITSE*, entre otros.

Con vista a evaluar tanto el desempeño del sistema de control, como la efectividad de los algoritmos de optimización, se ha realizado un estudio comparativo de cada uno de los diferentes ajustes realizados. Durante el estudio comparativo se ha evaluado el cumplimiento de los objetivos de control en el proceso bajo estudio. Uno de los objetivos es mantener constante la fuerza de corte, en operaciones de taladrado, frente a posibles perturbaciones que se produzcan debido al incremento de la profundidad de corte. El otro objetivo es evitar que se originen oscilaciones en la respuesta temporal de la fuerza de corte, lo cual resulta imprescindible para la protección de la herramienta de corte y por tanto el incremento de su vida útil. El cumplimiento del primero de los objetivos de control se ha cuantificado a partir del comportamiento del error de la variable controlada respecto a la variable de referencia, mediante su valor medio absoluto y cuadrático, ponderado y no ponderado por el tiempo (*IAE*, *ITAE*, *ISE*, *ITSE* e *ISTSE*). Por otra parte, el cumplimiento del otro objetivo se ha verificado a partir del valor del máximo sobrepaso (con respecto a la variable de referencia) de la respuesta temporal de la variable controlada.

Los resultados obtenidos en este capítulo, constituyen la base para el desarrollo futuro de sistemas de control inteligente con capacidades cognitivas, que incorporen estrategias de

competición, para ser utilizados en el control de procesos físicos complejos, con un comportamiento dinámico inherentemente no lineal y variables en el tiempo.

---

## CONCLUSIONES

---

Esta Tesis Doctoral basada en una metodología científica en la que interactúan métodos de investigación teórica y experimental, ha presentado el diseño, desarrollo e implementación de sistemas de control inteligente para la optimización de procesos complejos. El diseño está basado en diferentes estrategias de hibridación de técnicas del *soft-computing*, como son la computación borrosa, la neuro-computación y los métodos metaheurísticos. Los sistemas de control desarrollados han sido aplicados al proceso de taladrado de alto rendimiento, un proceso de fabricación que tiene un gran impacto en la producción de bienes y servicios.

Como paso inicial e indispensable de la metodología científica empleada, se ha realizado una revisión del estado de la ciencia y la técnica en los paradigmas elegidos dentro del amplio universo que ponen a nuestra disposición la Teoría General de Sistemas, la Teoría de Control, las técnicas de Inteligencia Artificial y las Ciencias de la Computación. Se ha prestado principal interés en los paradigmas con una fundamentación teórica sólida en el campo de la Inteligencia Artificial (IA), con un enfoque desde la perspectiva de la Automática y la Teoría de Sistemas. Es decir, el estudio se ha centrado en estrategias susceptibles de ser utilizadas en el modelado, monitorización y control de procesos físicos. Paradigmas que aprovechan tanto las ventajas de las técnicas básicas de Inteligencia Artificial, como las sinergias entre estas. De modo que permiten el diseño de sistemas de control cualitativamente superiores y capaces de ofrecer altas prestaciones en la optimización de procesos físicos complejos. Por este motivo la revisión se ha enfocado hacia las técnicas del *soft-computing*, analizando su evolución desde sus inicios hasta las tendencias actuales.

Tras el estudio, de la literatura revisada resalta, principalmente, la aplicabilidad de los sistemas de inferencia borrosa como un marco computacional eficiente para representar el conocimiento de expertos y, por tanto, viable en el control de procesos físicos complejos. Sin embargo, las tendencias más actuales apuntan hacia su hibridación con otras técnicas, como la

neuro-computación y los métodos metaheurísticos. De este modo se pueden diseñar sistemas inteligentes para el control con una representatividad más eficiente del conocimiento, con diseños relativamente sencillos y de bajo coste. Estos argumentos sustentaron la hipótesis de considerar métodos de optimización tales como los algoritmos genéticos, el algoritmo de temple simulado, el algoritmo de Nelder-Mead y el algoritmo de entropía cruzada, para el ajuste ó sintonía óptima de sistemas de inferencia borrosa. Esto ha permitido, además, examinar la potencialidad de la hibridación de estos sistemas con las redes neuronales adaptativas para el control de procesos.

Debido a que estas estrategias han sido aplicadas al control de procesos electro-mecánicos complejos, como es el caso de los procesos de mecanizado, se ha realizado, además, una breve revisión del estado de la técnica de estos procesos. Se ha presentado de forma breve como ha sido su evolución hasta nuestros días y las tendencias actuales en la aplicación de técnicas de IA desde la perspectiva del control, para hacer frente a la complejidad inherente de estos procesos y de esta forma mejorar su eficiencia.

Otro resultado de este trabajo constituye la obtención de una representación computacional eficiente y simple de procesos de elevada complejidad. Se han empleado tanto métodos teóricos como experimentales para el modelado de procesos de fabricación de alta complejidad, en específico el proceso de fresado de alta velocidad y el proceso de taladrado de alto rendimiento. Los modelos desarrollados han sido implementados en un lenguaje de alto nivel como Matlab/Simulink. Por una parte, a partir de modelos basados en ecuaciones físico-matemáticas, **se ha implementado un modelo computacional múltiple entrada-múltiple salida, para la estimación de las fuerzas de corte instantáneas en el proceso de fresado de alta velocidad.** Por otra parte, mediante la aplicación de técnicas de identificación experimental de sistemas dinámicos, **se ha obtenido e implementado un modelo múltiple entrada-simple salida, también del proceso de fresado de alta velocidad,** el cual tiene como entradas variables representativas de este proceso, como es el caso de la profundidad de corte, la velocidad de giro de la herramienta y su velocidad de avance y como salida la fuerza de corte resultante promedio por cada vuelta de la herramienta. También mediante identificación experimental, **se ha desarrollado un modelo simple entrada-simple salida del comportamiento dinámico de la fuerzas de corte, frente a cambios en la velocidad de avance de la herramienta, en el proceso de taladrado de alto rendimiento.**

**A través de un estudio comparativo con datos obtenidos experimentalmente, se ha verificado la efectividad de los modelos desarrollados. Estos modelos computacionales son contribuciones de esta Tesis Doctoral.** Representan, además, la base para el desarrollo de funciones de coste, primordiales dentro de la metodología de diseño propuesta para el ajuste óptimo de los sistemas de control inteligente.

Con vistas a profundizar en el conocimiento de las técnicas consideradas en esta Tesis Doctoral para el diseño de sistemas de control de procesos complejos, se ha realizado una revisión y estudio que trata algunas ideas conceptuales esenciales, aunque las estrategias de análisis dinámico de sistemas no han sido consideradas. Entre la amalgama de técnicas se han seleccionado solo algunas de las que tienen un consolidado cuerpo de doctrinas e impacto científico y técnico.

Como técnicas para el control de procesos complejos, en el estudio se han expuesto las ideas conceptuales y básicas de estrategias de control clásicas lineales, como el algoritmo de control proporcional, integral y derivativo, de muy amplia utilización. Además, se han considerado estrategias basadas en el *soft-computing*, como es el caso de los sistemas de inferencia borrosa y su hibridación con otras técnicas de inteligencia artificial, como las redes neuronales artificiales. De esta forma se combina, el estilo de razonamiento humano de los sistemas borrosos, con la estructura conexionista y los algoritmos de aprendizaje de las redes neuronales, con vistas a una mejoría del desempeño de los sistemas borrosos en la monitorización y control de sistemas complejos.

También como parte del estudio, se ha realizado un acercamiento científico a algunos métodos de optimización estocásticos y libres de gradiente, para su aplicación en el ajuste óptimo de las estrategias de control de procesos complejos, en específico de los sistemas de inferencia borrosa. El estudio se ha centrado en métodos de inspiración biológica, como es el caso de los algoritmos genéticos, métodos de inspiración física que se basan, principalmente, en la simulación de procesos físicos, como es el caso del algoritmo de temple simulado (*Simulated Annealing*) y el algoritmo de entropía cruzada (*Cross Entropy*) y métodos de inspiración geométrica, como el algoritmo de búsqueda directa de Nelder-Mead. Un estudio exhaustivo de los métodos de optimización accesibles en la literatura científica, está fuera del ámbito de esta Tesis Doctoral. Solo se han considerado algunos métodos representativos desde la perspectiva de la Automática.

El estudio de técnicas clásicas y de *soft-computing* y de algunos métodos de optimización, ha sido la base para el diseño y posterior implementación de sistemas de control optimizados, para su aplicación a un proceso de fabricación que se propone en la Tesis Doctoral. En específico se ha considerado el control de la fuerza de corte en el proceso de taladrado de alto rendimiento, como caso real de proceso electro-mecánico complejo. Este proceso es no lineal y variable en el tiempo. Una problemática es que durante su ejecución, a medida que se incrementa la profundidad de penetración de la herramienta en la pieza de trabajo, se dificulta el proceso de evacuación del material eliminado, lo cual conlleva un incremento de la temperatura, fuerza y pares de corte en la punta de la herramienta, induciendo por tanto, una disminución en su vida útil y una demanda de mayor potencia de la máquina que realiza la operación de taladrado de alto rendimiento. Este fenómeno se refleja en el comportamiento dinámico de las fuerzas de corte que se generan durante la ejecución del proceso. Así, **para el**

**diseño de los sistemas de control se ha planteado como objetivo de control, mantener constante la fuerza de corte ante el incremento en la profundidad de penetración de la herramienta, actuando de forma óptima sobre su velocidad de avance.**

Según la metodología de diseño, inicialmente, el ajuste de las estrategias de control se realiza mediante simulación del comportamiento temporal del proceso, el cual se representa mediante el modelo dinámico computacional eficiente del proceso de taladrado de alto rendimiento obtenido con anterioridad. Como paso siguiente y necesario, estas estrategias son validadas a través de su implementación en tiempo real en una plataforma experimental formada, principalmente, por un centro de mecanizado de alto rendimiento y controlando directamente la fuerza de corte en operaciones de taladrado de alto rendimiento.

### *Comentarios finales sobre el diseño de la estrategia de control lineal*

Como estrategia de control clásica lineal, **se ha implementado un algoritmo de control proporcional, integral y derivativo (PID)**, el cual se ha ajustado mediante el método clásico de Ziegler-Nichols por medio de un modelo dinámico del proceso de taladrado de alto rendimiento. **El desempeño de la estrategia de control ha sido corroborado experimentalmente, mediante operaciones de taladrado controladas a lazo cerrado y en tiempo real**, utilizando como interfaz de acceso al proceso la herramienta de Matlab, Real-Time Workshop (RTW). Se utilizaron tres condiciones de operación diferentes: el proceso de taladrado controlado a lazo cerrado con un filtro pasa bajo en la variable de referencia del sistema de control, también controlado a lazo cerrado pero sin filtrado en la referencia de la fuerza de corte y el proceso de taladrado a lazo abierto, es decir, sin el control PID activado.

Los experimentos realizados reflejan que **la estrategia de control con filtrado del valor de referencia mejora el comportamiento dinámico de la fuerza de corte, obteniéndose una respuesta del sistema de control sin sobrepaso, lo que no ocurre cuando se utiliza la estrategia de control PID básica**, es decir, sin la etapa de filtrado. No obstante, **en los dos casos el sistema de control es capaz de mantener la fuerza con muy poca variación alrededor de su valor de referencia, durante toda la operación de taladrado**, hecho que no se produce sin el control activado, es decir a lazo abierto, en el cual la fuerza de corte se incrementa en el tiempo, a medida que aumenta la profundidad durante el corte.

**El desempeño de la estrategia de control, se ha evaluado, además, a través de tres criterios de error o índices de desempeño** basados en el error a estado estacionario de la fuerza de corte respecto a su valor de referencia, como son: la integral del error absoluto (IAE), la integral del error absoluto ponderado en el tiempo (ITAE) y la integral del error cuadrático ponderado en el tiempo (ITSE). **Los índices obtenidos han mostrado la ventaja de aplicar una estrategia de control lineal PID a lazo cerrado**. Por otra parte, **la mejora de la operación de fabricación, se refleja en el aumento de la vida útil de la herramienta de**

corte, debido al mantenimiento casi constante de la fuerza de corte durante la operación de taladrado, y en la reducción que se produce en el tiempo de ejecución de la operación.

### *Comentarios finales sobre el diseño de los sistemas de control basados en soft-computing*

En el caso de la estrategia de control borrosa, se ha considerado un sistema de tipo **Mamdani** de dos entradas (la señal de error en la fuerza de corte y su variación) y una salida (el cambio en la velocidad de avance en el proceso de taladrado). Se han utilizado 49 reglas en su sistema de inferencia y el universo de discurso de cada una de las variables se divide en 7 particiones borrosas distribuidas uniformemente y caracterizadas por funciones de pertenencia de tipo triangular. La construcción de la base de reglas se ha realizado a partir de la plantilla de MacVicar-Whelan, que es considerada como el sentido común con mayor coincidencia entre ingenieros con experiencia en control borroso.

Se ha planteado como objetivo de control, el mantenimiento de la fuerza de corte constante durante toda la operación de taladrado, a pesar de la presencia de perturbaciones debido al incremento de la profundidad de corte, modificando la velocidad de avance de la herramienta de corte. Para dar cumplimiento a este objetivo, el algoritmo de control borroso se ha ajustado mediante la selección o ajuste óptimo de los factores de escala de sus entradas. Para ello, se utilizaron los métodos de optimización estocásticos y libres de gradiente estudiados previamente (Nelder-Mead, algoritmos genéticos, temple simulado y entropía cruzada). Para cada uno de los sistemas de control borrosos ajustados por las diferentes estrategias de optimización, se ha seguido una metodología de diseño que consta básicamente de las siguientes etapas:

1. Selección de los valores iniciales de los factores de escala o parámetros del controlador.
2. A partir de una función de coste, basada en el error y que utiliza como variables de entrada los factores de escala, se ha ejecutado el algoritmo de optimización, mediante la simulación del proceso de taladrado representado por su modelo dinámico computacional. El procedimiento de optimización se detiene hasta alcanzar un valor mínimo de la función de coste o si se satisfacen condiciones de paradas.
3. Simulación del sistema de control, comparando su desempeño con los parámetros ajustados inicialmente y los obtenidos tras el procedimiento de optimización.
4. Debido a que las acciones de control se envían a través de una arquitectura distribuida, se evalúa la influencia de los retardos de las comunicaciones en el desempeño de controlador, lo cual se simula mediante la introducción de retardos temporales, aleatorios, uniformemente distribuidos y acotados, en el modelo computacional del proceso. Comprobándose y evaluándose el comportamiento del sistema de control a partir de los resultados de simulación.

5. Implementación experimental del sistema de control, controlando en tiempo real operaciones de taladrado.
6. Comprobación del cumplimiento de los objetivos de control y validación del comportamiento del sistema de control a partir del cálculo de cifras de mérito.

**En todos los casos, para el ajuste óptimo del sistema de control borroso, se ha utilizado como función objetivo a minimizar la integral del error absoluto ponderado en el tiempo (ITAE), constituyendo, desde el punto de vista científico y técnico, una de las aportaciones de este trabajo.** Este criterio o cifra de mérito es muy interesante ya que penaliza las respuestas transitorias temporales de larga duración y es muy sensible ante variaciones de parámetros, en comparación con otros criterios como la integral del error absoluto (IAE) o la integral del error cuadrático ponderado en el tiempo (ITSE), entre otros.

Además de las estrategias de control borrosas ajustadas mediante técnicas metaheurísticas, **se ha diseñado e implementado un sistema de control por modelo interno, basado en modelos híbridos neuro-borrosos** de tipo ANFIS (*Adaptive Network based Fuzzy Inference System*), aplicado también a la regulación automática a lazo cerrado de la fuerza de corte en el proceso de taladrado de alto rendimiento.

El esquema de control por modelo interno está formado por un modelo directo y uno inverso empleando, en este caso para los dos, sistemas de inferencia borrosa de tipo Takagi-Sugeno-Kang (TSK), ajustados mediante redes neuronales adaptativas. Ambos modelos tienen una entrada y una salida. Para el caso del modelo directo, la velocidad de avance de la herramienta de corte es la variable de entrada y la fuerza de corte la variable de salida. Por el contrario, en el caso del modelo inverso, a partir de una determinada fuerza de corte como entrada, calcula como salida la correspondiente velocidad de avance. Los modelos están formados por 2 reglas en su sistema de inferencia, el universo de discurso de sus correspondientes variables de entrada, se divide por 2 particiones borrosas caracterizadas por funciones de pertenencia de tipo campana generalizada y funciones polinómicas de primer orden en el consecuente de cada regla.

La estructura de control fue diseñada de forma similar a la metodología descrita anteriormente, con la diferencia de que, en el segundo paso, ambos modelos fueron entrenados a partir de datos obtenidos experimentalmente de operaciones de taladrado controladas a lazo cerrado. Además, para el ajuste de los modelos se utiliza una estrategia de entrenamiento híbrida, primeramente el consecuente de cada regla, en este caso los polinomios de primer orden, se estiman por medio del método de mínimos cuadrados. Como segunda etapa del entrenamiento, se emplea el método de aprendizaje del gradiente descendiente y propagación hacia atrás de los errores para determinar el antecedente de las reglas, es decir para ajustar las funciones de pertenencias de las variables de entrada. El entrenamiento se realiza de forma iterativa hasta que se cumpla cualquiera de las condiciones



de parada, por ejemplo que el error de salida del modelo esté por debajo del valor deseado o se sobrepase un número de iteraciones máximas.

Una vez diseñadas y aplicadas cada una de las estrategias de control, como parte de la metodología científica de la Tesis Doctoral y también como se ha comentado en los pasos de la metodología de diseño, todas las estrategias desarrolladas han sido validadas rigurosamente en una plataforma experimental utilizada como soporte tecnológico. **La validación de las estrategias mediante su implementación en tiempo real, actuando directamente sobre operaciones de taladrado de alto rendimiento, ha permitido corroborar los resultados obtenidos en simulación, así como la efectividad de los métodos metaheurísticos empleados en el ajuste de los sistemas de control inteligente.**

Los estudios comparativos mostraron que, **en el caso de la aplicación de las técnicas metaheurísticas para el ajuste del sistema de control borroso de 49 reglas, las respuestas temporales de los sistemas de control, obtenidas experimentalmente son similares entre ellas y a las obtenidas en simulación.** Esto se debe a que **en todos los casos se han alcanzado parámetros óptimos o sub-óptimos de los sistemas de control borrosos.** A pesar de su similitud, **el desempeño en general del sistema de control optimizado mediante el método de entropía cruzada, es ligeramente superior al resto de técnicas utilizadas, además de ofrecer una muy buena relación tiempo de cómputo y total de parámetros de ajuste del método requeridos.**

**El algoritmo de entropía cruzada ofrece un tiempo de cómputo menor que el método de temple simulado y menor que los algoritmos genéticos, y por otra parte, tiene menos parámetros de ajuste que el método de algoritmos genéticos.** Sin embargo, ha de destacarse que la comparativa basada en el tiempo de cómputo puede resultar un tanto engañosa, porque no se ha tenido en cuenta en la comparación, por ejemplo, el tiempo total empleado, durante todo el procedimiento de ajuste mediante cada estrategia de optimización, para obtener el mejor conjunto sub-óptimo de los parámetros del controlador (factores de escala).

Además, la validación de las estrategias mediante su implementación en tiempo real, también ha dado la posibilidad de realizar estudios comparativos entre las estrategias basadas en técnicas de *soft-computing*, las estrategias lineales y el proceso de taladrado a lazo de control abierto. Los estudios comparativos se han basado en la cuantificación del cumplimiento de los objetivos de control, a partir de cifras de méritos que caracterizan el comportamiento dinámico de sistemas de control, como el máximo sobrepaso de la fuerza de corte (variable controlada del sistema de control) y los valores medios de su error a estado estacionario respecto a su valor deseado (ej., índices de desempeño *ITAE*, *IAE*, *ITSE*, entre otros).

Las cifras de mérito han corroborado que de las estrategias diseñadas, las basadas en técnicas del *soft-computing*, presentan un desempeño superior a las estrategias lineales, al menos controlando operaciones de taladrado de alto rendimiento a lazo cerrado en fase de simulación y en tiempo real. Esto hace que las estrategias de control no lineales, basadas en técnicas de Inteligencia Artificial, puedan ser consideradas más efectivas para hacer frente a la complejidad intrínseca de procesos físicos, como es el caso de los procesos de fabricación estudiados en esta Tesis Doctoral, y conducen, por tanto, a una mejoría sustancial de su funcionamiento y eficiencia.

Las estrategias computacionales diseñadas e implementadas, tanto basadas en técnicas de control lineal como en técnicas de *soft-computing*, aplicadas a la optimización del proceso de taladrado de alto rendimiento, constituyen, desde el punto de vista científico y técnico, otras de las aportaciones de este trabajo. De forma resumida, las contribuciones científico-técnicas realizadas en esta Tesis Doctoral han sido:

1. El diseño e implementación, en un lenguaje de alto rendimiento, de un modelo computacional que emula el comportamiento de variables representativas en procesos de fabricación. Análisis del caso particular de las fuerzas de corte en el proceso de fresado de alta velocidad.
2. El diseño e implementación de un modelo dinámico computacionalmente eficiente de un proceso de fabricación complejo. Aplicación al proceso de taladrado de alto rendimiento.
3. El diseño e implementación de un modelo dinámico múltiple entrada-simple salida, computacionalmente eficiente, de un proceso de fabricación complejo. Aplicación al caso particular del proceso de fresado de alta velocidad.
4. El diseño e implementación de una estrategia de control lineal para la optimización en tiempo real de un proceso de fabricación. Aplicación al caso particular del proceso de taladrado de alto rendimiento.
5. El diseño e implementación de una estrategia de control por modelo interno, basada en modelos híbridos neuro-borrosos, para la optimización en tiempo real de un proceso de fabricación complejo. Aplicación al proceso de taladrado de alto rendimiento.
6. El diseño e implementación de estrategias de control basadas en la hibridación entre los sistemas de inferencia borrosa y métodos metaheurísticos, con vistas a optimizar un proceso electromecánico complejo. En específico métodos de inspiración biológica como los algoritmos genéticos, métodos basados en la simulación de procesos físicos como el algoritmo de temple simulado y el algoritmo de entropía cruzada y métodos de búsqueda directa como el algoritmo de Nelder-Mead. Aplicación de estas

estrategias al caso particular del control en tiempo real del proceso de taladrado de alto rendimiento.

7. El desarrollo de un estudio comparativo entre técnicas de *soft-computing* y estrategias lineales para el control en tiempo real de un proceso de fabricación complejo.
8. El desarrollo de un estudio comparativo de los resultados obtenidos de la aplicación de las técnicas de optimización al ajuste de un sistema de inferencia borrosa aplicado al control de un proceso de fabricación complejo.

## Lista de publicaciones

Algunas de las contribuciones científicas y técnicas realizadas durante toda la etapa de investigación de esta Tesis Doctoral, han sido puestas a consideración por la comunidad científica en diferentes publicaciones realizadas, ya sea en artículos de revistas indexadas (SCI, *Science Citation Index*), actas de congresos científicos o en patentes solicitadas, las cuales se relacionan a continuación.

Artículos en revistas SCI:

- R. E. Haber, A. Gajate, S. Y. Liang, R. Haber-Haber, and **R. M. del Toro**, "An Optimal Fuzzy Controller for a High-Performance Drilling Process Implemented over an Industrial Network," *International Journal of Innovative Computing, Information and Control*, vol. 7, March 2011 (In Press).
- A. Gajate, R. Haber, **R. del Toro**, P. Vega, and A. Bustillo, "Tool wear monitoring using neuro-fuzzy techniques: a comparative study in a turning process," *Journal of Intelligent Manufacturing*, pp. 1-14, 2010.
- R. E. Haber, **R. M. del Toro**, and A. Gajate, "Optimal fuzzy control system using the cross-entropy method. A case study of a drilling process," *Information Sciences*, vol. 180, pp. 2777-2792, 2010.
- A. Gajate, R. E. Haber, and **R. M. del Toro**, "Neurofuzzy Drilling Force-Based Control in an Ethernet-Based Application," *International Journal of Innovative Computing, Information and Control*, vol. 6, pp. 373-386, January 2010.
- **R. M. del Toro-Matamoros**, R. E. Haber, J. Pamies, and A. Bustillo, "Modelling and simulation of the high-speed milling process based on MATLAB-SIMULINK," *Revista de Metalurgia*, vol. 44, pp. 176-188, 2008.

- R. Haber-Haber, R. Haber, M. Schmittiel, and **R. M. del Toro**, "A classic solution for the control of a high-performance drilling process," *International Journal of Machine Tools and Manufacture*, vol. 47, pp. 2290-2297, 2007.

#### Artículos en actas de congresos:

- R. E. Haber, R. Haber-Haber, **R. M. del Toro**, and J. R. Alique, "Using simulated annealing for optimal tuning of a PID controller for time-delay systems. An application to a high-performance drilling process," in *Lecture Notes in Computer Science (including sub series Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. vol. 4507 LNCS San Sebastian, 2007, pp. 1155-1162.
- **R. del Toro**, M. Schmittiel, R. Haber-Guerra and R. Haber-Haber, "System identification of the high performance drilling process for network-based control," in *Proc. of ASME IDETC 2007: 21st. Biennial Conference on Mechanical Vibration and Noise (VIB)* Las Vegas, NV, USA, 2007.

#### Patentes solicitadas:

- R. E. Haber Guerra, F. Martínez Puente, **R. M. del Toro**, and B. Caballero Retamosa, "Fuzzy logic-based control methods for drilling". Consejo Superior de Investigaciones Científicas, Spain. **PCT/ES2008/070039**, 2008 (Pendiente).
- R. E. Haber Guerra, F. Martínez Puente, **R. M. del Toro**, M. C. Schmittiel, and D. Martín Andrés. "Modelling method and system for drilling processes". Consejo Superior de Investigaciones Científicas, Spain. **PCT/ES2008/070038**, 2008 (Pendiente).

## Trabajo futuro

El desarrollo de esta Tesis Doctoral ha sido fruto de trabajos de investigación enfocados, principalmente, en el campo de los sistemas de control inteligente y su aplicación a la optimización de procesos complejos, basándose fundamentalmente en el desarrollo de estrategias de hibridación de técnicas del *soft-computing*.

A pesar de los resultados alcanzados y las contribuciones realizadas, esta disertación sólo constituye una etapa dentro de una línea de investigación más amplia, cuyo objetivo principal apunta hacia el desarrollo de un método para el diseño de un sistema de control cognitivo artificial, basado en modelos internos, para procesos de mecanizado de alto rendimiento. Desde el punto de vista científico, el objetivo es evolucionar los controladores inteligentes a

partir de la imitación de la evolución de la mente biológica, incorporando estrategias de competición como aspecto clave en los sistemas cognitivos. En específico, se pretende dotar a estos sistemas de capacidades *self*-\* (auto-optimización, auto-adaptación, auto-configuración y auto-aprendizaje) más eficientes.

Como paso siguiente las estrategias de optimización de mayor eficiencia computacional, serán integradas en sistemas intermediarios con especificaciones de tiempo real. **Esto permitirá el desarrollo de unidades neuro-borrosas auto-optimizadas dotadas de razonamiento inductivo y transductivo.**

El desarrollo de estas unidades puede estar basado en el diseño de estrategias de control borrosas que aprovechen las capacidades, tanto de las redes neuronales artificiales, como de las técnicas metaheurísticas. Como se ha comentado en la literatura con anterioridad y se ha demostrado en el desarrollo de este trabajo, el empleo de técnicas metaheurísticas para ajustar los parámetros (ej., factores de escala) y la utilización de las redes neuronales adaptativas para ajustar las funciones de pertenencia, causando efectos “macroscópicos” y “mesoscópicos” en la base de reglas, puede contribuir a una mejora de la eficiencia en la representación del conocimiento, debido a que podría lograrse un mayor refinamiento en el ajuste de la base de conocimientos de los sistemas de inferencia borrosa.



---

## BIBLIOGRAFÍA

---

- [1] E. Commission, "MANUFUTURE: A vision for 2020, Assuring the future of manufacturing in Europe," *Office for Official Publications of the EC*, p. 20 2004.
- [2] "Sistema de control cognitivo en red para procesos de mecanizado de alto rendimiento (COGNETCON)," Ministerio de Ciencia e Innovación. DPI2008-01978, 2009 - 2011.
- [3] R. E. Haber, C. R. Peres, A. Alique, S. Ros, C. González, and J. R. Alique, "Toward intelligent machining: Hierarchical fuzzy control for the end milling process," *IEEE Transactions on Control Systems Technology*, vol. 6, pp. 188-199, 1998.
- [4] L. A. Zadeh, "Soft computing and fuzzy logic," *IEEE Software*, vol. 11, pp. 48-56, 1994.
- [5] J. L. Verdegay, "De los conjuntos fuzzy a la Soft Computing," *Ágora, Papeles de Filosofía*, vol. 24, pp. 29-48, 2005.
- [6] L. A. Zadeh, "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 2, pp. 23-25, 1998.
- [7] P. P. Bonissone, Y. U. T. O. Chen, K. Goebel, and P. S. Khedkar, "Hybrid soft computing systems: Industrial and commercial applications," *Proceedings of the IEEE*, vol. 87, pp. 1641-1667, 1999.
- [8] J. L. Verdegay, R. R. Yager, and P. P. Bonissone, "On heuristics as a fundamental constituent of soft computing," *Fuzzy Sets and Systems*, vol. 159, pp. 846-855, Apr 2008.
- [9] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers and Operations Research*, vol. 13, pp. 533-549, 1986.
- [10] I. H. Osman and G. Laporte, "Metaheuristics: A bibliography," *Annals of Operations Research*, vol. 63, pp. 513-623, 1996.

- [11] M. Dorigo and M. Birattari, "Swarm intelligence," *Scholarpedia*, vol. 2(9), 2007.
- [12] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial System*. New York: Oxford University Press, 1999.
- [13] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, pp. 29-41, 1996.
- [14] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE International Conference on Neural Networks - Conference Proceedings*, Perth, Aust, 1995, pp. 1942-1948.
- [15] T. Weise, "Global Optimization Algorithms. Theory and Application," <http://www.it-weise.de>, 2009.
- [16] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach.*, 2nd ed.: Prentice Hall, 2002.
- [17] V. Černý, "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, pp. 41-51, 1985.
- [18] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671-680, 1983.
- [19] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, vol. 13, pp. 533-549, 1986.
- [20] R. Y. Rubinstein, "Optimization of computer simulation models with rare events," *European Journal of Operational Research*, vol. 99, pp. 89-112, 1997.
- [21] J. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, pp. 308-313, 1965.
- [22] P. P. Bonissone, "Hybrid Soft Computing Systems: Where Are We Going?," in *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, Germany, 2000, pp. 739-746.
- [23] E. H. Mamdani and S. Assilian, "EXPERIMENT IN LINGUISTIC SYNTHESIS WITH A FUZZY LOGIC CONTROLLER," *International Journal of Man-Machine Studies*, vol. 7, pp. 1-13, 1975.
- [24] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, pp. 116-132, 1985.
- [25] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol. 28, pp. 15-33, 1988.
- [26] J. Moody and C. J. Darken, "Fast Learning in Networks of Locally-Tuned Processing Units," *Neural Computation*, vol. 1, pp. 281-294, 1989.
- [27] W. T. Miller Iii, F. H. Glanz, and L. G. Kraft Iii, "CMAC: An associative neural network alternative to backpropagation," *Proceedings of the IEEE*, vol. 78, pp. 1561-1567, 1990.



- 
- [28] R. R. Selmic and F. L. Lewis, "Neural-network approximation of piecewise continuous functions: Application to friction compensation," *IEEE Transactions on Neural Networks*, vol. 13, pp. 745-751, 2002.
  - [29] R. Griño, G. Cembrano, and C. Torras, "Nonlinear system identification using additive dynamic neural networks - two on-line approaches," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, pp. 150-165, 2000.
  - [30] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*: Prentice Hall, 1991.
  - [31] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, pp. 59-69, 1982.
  - [32] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, pp. 2554-2558, 1982.
  - [33] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Computer Vision, Graphics and Image Processing*, vol. 37, pp. 54-115, 1987.
  - [34] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735-1780, 1997.
  - [35] J. Zhao and J. Shawe-Taylor, "Stochastic connection neural networks," in *Artificial Neural Networks, 1995., Fourth International Conference on*, 1995, pp. 35-39.
  - [36] D. E. Goldberg, *Genetic Algorithms in Search, Optimization & Machine Learning*: Addison-Wesley Publishing Co., 1989.
  - [37] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*, 4th ed.: The MIT press, 1992.
  - [38] L. J. Fogel, "On the organization of intellect," Los Angeles, California, USA: UCLA University of California, 1964.
  - [39] I. Rechenberg, *Cybernetic Solution Path of an Experimental Problem*: Royal Aircraft Establishment. Translation No. 1122, 1965.
  - [40] L. A. Zadeh, "Probability measures of Fuzzy events," *Journal of Mathematical Analysis and Applications*, vol. 23, pp. 421-427, 1968.
  - [41] P. Smets, "The degree of belief in a fuzzy event," *Information Sciences*, vol. 25, pp. 1-19, 1981.
  - [42] D. J. Montana and L. Davis, "Training feedforward neural networks using genetic algorithms ." in *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI-89)*, San Francisco, CA, 1989, pp. 762-767.
  - [43] D. B. Fogel, L. J. Fogel, and V. W. Porto, "Evolutionary methods for training neural networks," in *IEEE Conference on Neural Networks for Ocean Engineering*, Washington, DC, USA, 1991, pp. 317-327.
  - [44] J. R. McDonnell and D. Waagen, "Evolving neural network connectivity," in *IEEE International Conference on Neural Networks (ICNN'93)*, San Francisco, CA, 1993, pp. 863-868.
-

- [45] V. Maniezzo, "Genetic evolution of the topology and weight distribution of neural networks," *IEEE Transactions on Neural Networks*, vol. 5, pp. 39-53, 1994.
- [46] R. A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 1, pp. 295-307, 1988.
- [47] P. Arabshahi, J. J. Choi, R. J. Marks, and T. P. Caudell, "Fuzzy control of backpropagation," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'92)*, San Diego, CA, USA, 1992, pp. 967-972.
- [48] F. Herrera and M. Lozano, "Adaptive genetic algorithms based on fuzzy techniques," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'96)*, Granada, Spain, 1996, pp. 775-780.
- [49] R. Kruse, "Fuzzy neural network," *Scholarpedia*, vol. 3(11), 2008.
- [50] D. Nauck, "A Fuzzy Perceptron as a Generic Model for Neuro-Fuzzy Approaches," in *Proceedings of Fuzzy-systeme'94, Second GI-workshop*, Munich, 1994.
- [51] D. Nauck and R. Kruse, "Neuro-Fuzzy Classification with NEFCLASS," in *Operations Research Proceedings 1995*, Berlin, 1996, pp. 294-299.
- [52] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, pp. 665-685, 1993.
- [53] J. S. R. Jang and C. T. Sun, "Functional equivalence between radial basis function networks and fuzzy inference systems," *Neural Networks, IEEE Transactions on*, vol. 4, pp. 156-159, 1993.
- [54] H. Bersini, G. Bontempi, and C. Decaestecker, "Comparing RBF and fuzzy inference systems on theoretical and practical basis," *Proc. Int. Conf. Artificial Neural Networks. ICANN '95*, vol. 1, pp. 169-174, 1995.
- [55] O. Cordón, F. Herrera, and M. Lozano, "A Classified Review On The Combination Fuzzy Logic-Genetic Algorithms Bibliography: 1989 -1995," Dept. of Computer Science and A.I., University of Granada. Technical Report DECSAI-95129, December 1996.
- [56] C. L. Karr, "Design of an adaptive fuzzy logic controller using genetic algorithms," in *Proc. Int. Conf. Genetic Algorithms (ICGA'91)*, San Diego, CA, 1991, pp. 450-456.
- [57] M. A. Lee and H. Tagaki, "Dynamic control of genetic algorithm using fuzzy logic techniques," in *Proc. 5th Int. Conf. Genetic Algorithms.*, San Mateo, CA, 1992, pp. 76-83.
- [58] P. P. Bonissone, P. S. Khedkar, and Y. Chen, "Genetic algorithms for automated tuning of fuzzy controllers: a transportation application," in *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'96)*, New Orleans, LA, 1996, pp. 674-680.
- [59] L. Zheng, "A practical guide to tune of proportional and integral (PI) like fuzzy controllers," in *IEEE International Conference on Fuzzy Systems*, 1992, pp. 633-640.
- [60] M. Correa, C. Bielza, M. D. J. Ramirez, and J. R. Alique, "A Bayesian network model for surface roughness prediction in the machining process," *International Journal of Systems Science*, vol. 39, pp. 1181-1192, 2008.

- 
- [61] E. B. Flynn and M. D. Todd, "Bayesian probabilistic structural modeling for optimal sensor placement in ultrasonic guided wave-based structural health monitoring," in *Proceedings of SPIE - The International Society for Optical Engineering*, San Diego, CA, 2010.
- [62] P. Elliott and B. Williams, "DNNF-based belief state estimation," in *Proceedings of the National Conference on Artificial Intelligence*, Boston, MA, 2006, pp. 36-41.
- [63] C. Alippi and V. Piuri, "Experimental neural networks for prediction and identification," *IEEE Transactions on Instrumentation and Measurement*, vol. 45, pp. 670-676, 1996.
- [64] D. W. Coit, B. T. Jackson, and A. E. Smith, "Static neural network process models: Considerations and case studies," *International Journal of Production Research*, vol. 36, pp. 2953-2967, 1998.
- [65] C. Feng, X. Wang, and Z. Yu, "Neural networks modelling of honing surface roughness parameter defined by ISO 13565," *SIAM Journal of Manufacturing Systems*, vol. 21, pp. 1-35, 2002.
- [66] C. Barnes, S. Brown, G. Flake, R. Jones, M. O. Rourke, and Y. C. Lee, "Applications of neural networks to process control and modelling," *Artificial Neural Networks, Proceedings of 1991 Internal. Conf. Artif. Neur. Nets*, vol. 1, pp. 321-326, 1991.
- [67] K. J. Hunt, D. Sbarbaro, R. Zbikowski, and P. J. Gawthrop, "Neural networks for control systems - A survey," *Automatica*, vol. 28, pp. 1083-1112, 1992.
- [68] L. Jin, P. N. Nikiforuk, and M. M. Gupta, "Dynamic recurrent neural networks for control of unknown nonlinear systems," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 116, pp. 567-576, 1994.
- [69] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 1, pp. 4-27, 1990.
- [70] R. Babuska and H. B. Verbruggen, "An overview of fuzzy modeling for control," *Control Engineering Practice*, vol. 4, pp. 1593-1606, 1996.
- [71] R. E. Haber, A. Alique, J. R. Alique, R. Haber-Haber, and S. Ros, "Current trends and future developments of new control systems based on fuzzy logic and their application to high speed machining," *Revista de Metalurgia (Madrid)*, vol. 38, pp. 124-133, 2002.
- [72] M. Santos, R. López, and J. M. De La Cruz, "A neuro-fuzzy approach to fast ferry vertical motion modelling," *Engineering Applications of Artificial Intelligence*, vol. 19, pp. 313-321, 2006.
- [73] M. A. Denai, F. Palis, and A. Zeghib, "Modeling and control of non-linear systems using soft computing techniques," *Applied Soft Computing Journal*, vol. 7, pp. 728-738, 2007.
- [74] F. C. Sun, Z. Q. Sun, and H. X. Li, "Stable adaptive controller design of robotic manipulators via neuro-fuzzy dynamic inversion," *Journal of Robotic Systems*, vol. 22, pp. 809-819, 2005.
-

- [75] A. M. Shahri, B. J. Evans, and F. Naghdy, "Neuro-fuzzy adaptive torque control of a SCARA robot," in *Proceedings of the Australian and New Zealand Conference on Intelligent Information Systems*, Adelaide, Aust, 1996, pp. 241-244.
- [76] X. J. Liu, F. Lara-Rosano, and C. W. Chan, "Model-reference adaptive control based on neurofuzzy networks," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 34, pp. 302-309, 2004.
- [77] R. E. Haber, J. R. Alique, A. Alique, and R. H. Haber, "Controlling a complex electromechanical process on the basis of a neurofuzzy approach," *Future Generation Computer Systems*, vol. 21, pp. 1083-1095, 2005.
- [78] W. A. Farag, V. H. Quintana, and G. Lambert-Torres, "A genetic-based neuro-fuzzy approach for modeling and control of dynamical systems," *IEEE Transactions on Neural Networks*, vol. 9, pp. 756-767, 1998.
- [79] B. Carse, T. C. Fogarty, and A. Munro, "Evolving fuzzy rule based controllers using genetic algorithms," *Fuzzy Sets and Systems*, vol. 80, pp. 273-293, 1996.
- [80] A. C. Nearchou, "Adaptive navigation of autonomous vehicles using evolutionary algorithms," *Artificial Intelligence in Engineering*, vol. 13, pp. 159-173, 1999.
- [81] D. Martín, R. Del Toro, R. Haber, and J. Dorronsoro, "Optimal tuning of a networked linear controller using a multi-objective genetic algorithm and its application to one complex electromechanical process," *International Journal of Innovative Computing, Information and Control*, vol. 5, pp. 3405-3414, 2009.
- [82] S. Y. Liang, R. L. Hecker, and R. G. Landers, "Machining process monitoring and control: The state-of-the-art," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 126, pp. 297-310, 2004.
- [83] R. Bueno, "Tendencias en los procesos de mecanizado," in *I Congreso Internacional de Diseño, Tecnologías e Ingeniería de Producto (Protodesign)* Valencia, Spain: <http://www.interempresas.net>, 2003.
- [84] "Remoción de Material: Mecanizado no Tradicional," Facultad de Ingeniería. Universidad Nacional de la Pampa. <http://www.ing.unlpam.edu.ar/~material/tecmechanica/material>.
- [85] E. P. DeGarmo, J. T. Black, and R. A. Kohser, *Materiales y procesos de fabricación*, 2nd ed. vol. 1. Barcelona: Reverte, S.A., 1994.
- [86] L. N. López de Lacalle Marcaide, J. A. Sánchez Galíndez, and A. Lamikiz Menchaca, *Mecanizado de alto rendimiento. Procesos de arranque*: Ediciones Técnicas Ízaro, 2004.
- [87] G. Byrne, D. Dornfeld, and B. Denkena, "Advancing cutting technology," *CIRP Annals - Manufacturing Technology*, vol. 52, pp. 483-507, 2003.
- [88] K. L. Petri, R. E. Billo, and B. Bidanda, "A Neural Network Process Model for Abrasive Flow Machining Operations," *Journal of Manufacturing Systems*, vol. 17, pp. 52-64, 1998.
- [89] H. C. Zhang and S. H. Huang, "Applications of neural networks in manufacturing: A state-of-the-art survey," *International Journal of Production Research*, vol. 33, pp. 705-728, 1995.

- 
- [90] Z. J. Viharos, L. Monostori, and S. Markos, "Selection of input and output variables of ANN based modeling of cutting processes," *Proceedings of the X. Workshop on Supervising and Diagnostics of Machining Systems of CIRP*, vol. 26, p. 121, 1999.
- [91] G. Sathyanarayanan, I. J. Lin, and M.-K. Chen, "Neural network modelling and multiobjective optimization of creep feed grinding of superalloys," *International Journal of Production Research*, vol. 30, pp. 2421-2438, 1992.
- [92] R. K. Jain, V. K. Jain, and P. K. Kalra, "Modelling of abrasive flow machining process: A neural network approach," *Wear*, vol. 231, pp. 242-248, 1999.
- [93] W. Grzesik and S. Brol, "Hybrid approach to surface roughness evaluation in multistage machining processes," *Journal of Materials Processing Technology*, vol. 134, pp. 265-272, 2003.
- [94] Y. C. Shin and P. Vishnupad, "Neuro-fuzzy control of complex manufacturing processes," *International Journal of Production Research*, vol. 34, pp. 3291-3309, 1996.
- [95] Y. T. Chen and S. R. T. Kumara, "Fuzzy logic and neural networks for design of process parameters: A grinding process application," *International Journal of Production Research*, vol. 36, pp. 395-415, 1998.
- [96] R. J. Kuo and P. H. Cohen, "Manufacturing process control through integration of neural networks and fuzzy model," *Fuzzy Sets and Systems*, vol. 98, pp. 15-31, 1998.
- [97] M. K. Kamatala, E. T. Baumgartner, and K. S. Moon, "Turned surface finish prediction based on fuzzy logic theory," *Proceedings of the 20th International Conference on Computer and Industrial Engineering*, vol. 1, pp. 101-104, 1996.
- [98] K. Hashmi, M. A. El Baradie, and M. Ryan, "Fuzzy logic based intelligent selection of machining parameters," *Computers and Industrial Engineering*, vol. 35, pp. 571-574, 1998.
- [99] M. Milfelner, J. Kopac, F. Cus, and U. Zuperl, "Genetic equation for the cutting force in ball-end milling," *Journal of Materials Processing Technology*, vol. 164-165, pp. 1554-1560, 2005.
- [100] T. M. A. Maksoud and M. R. Atia, "Review of intelligent grinding and dressing operations," *Machining Science and Technology*, vol. 8, pp. 263-276, 2004.
- [101] C. S. Rao and R. R. Srikant, "Tool wear monitoring - An intelligent approach," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 218, pp. 905-912, 2004.
- [102] P. Fu and A. D. Hope, "An intelligent signal feature pattern recognition architecture for condition monitoring of automatic machining processes," in *Proceedings - 2004 International Conference on Intelligent Mechatronics and Automation*, Chengdu, 2004, pp. 552-556.
- [103] Y. M. Ali and L. C. Zhang, "A fuzzy model for predicting burns in surface grinding of steel," *International Journal of Machine Tools and Manufacture*, vol. 44, pp. 563-571, 2004.
-

- [104] M. Liang, T. Yeap, and A. Hermansyah, "A fuzzy system for chatter suppression in end milling," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 218, pp. 403-417, 2004.
- [105] R. E. Haber and A. Alique, "Intelligent process supervision for predicting tool wear in machining processes," *Mechatronics*, vol. 13, pp. 825-849, 2003.
- [106] W. L. R. Ip, "A fuzzy basis material removal optimization strategy for sculptured surface machining using ball-nosed cutters," *International Journal of Production Research*, vol. 36, pp. 2553-2571, 1998.
- [107] Y. H. Lee, B. H. Yang, and K. S. Moon, "An economic machining process model using fuzzy non-linear programming and neural network," *International Journal of Production Research*, vol. 37, pp. 835-847, 1999.
- [108] H. Al-Wedyan, K. Demirli, and R. Bhat, "A technique for fuzzy logic modeling of machining process," in *Annual Conference of the North American Fuzzy Information Processing Society - NAFIPS*, Vancouver, BC, 2001, pp. 3021-3026.
- [109] G. G. Wang and S. Q. Xie, "Optimal process planning for a combined punch-and-laser cutting machine using ant colony optimization," *International Journal of Production Research*, vol. 43, pp. 2195-2216, 2005.
- [110] G. Skrabalak, M. Zybura-Skrabalak, and A. Ruszaj, "Building of rules base for fuzzy-logic control of the ECDM process," *Journal of Materials Processing Technology*, vol. 149, pp. 530-535, 2004.
- [111] A. Behrens and J. Ginzel, "Neuro-fuzzy process control system for sinking EDM," *Journal of Manufacturing Processes*, vol. 5, pp. 33-39, 2003.
- [112] X. Shi, P. Schillings, and D. Boyd, "Applying artificial neural networks and virtual experimental design to quality improvement of two industrial processes," *International Journal of Production Research*, vol. 42, pp. 101-118, 2004.
- [113] B. F. Yousef, G. K. Knopf, E. V. Bordatchev, and S. K. Nikumb, "Neural network modeling and analysis of the material removal process during laser machining," *International Journal of Advanced Manufacturing Technology*, vol. 22, pp. 41-53, 2003.
- [114] Z. C. Lin and C. Y. Liu, "Analysis and application of the adaptive neuro-fuzzy inference system in prediction of CMP machining parameters," *International Journal of Computer Applications in Technology*, vol. 17, pp. 80-89, 2003.
- [115] N. Ding, L. Wang, G. Li, D. Chen, X. Li, and D. Zhang, "Human-machine intelligent control system for grinding," *Chinese Journal of Mechanical Engineering (English Edition)*, vol. 17, pp. 26-29, 2004.
- [116] J. Balic, "Neural-network-based numerical control for milling machine," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 40, pp. 343-358, 2004.
- [117] R. J. Lian, B. F. Lin, and J. H. Huang, "A grey prediction fuzzy controller for constant cutting force in turning," *International Journal of Machine Tools and Manufacture*, vol. 45, pp. 1047-1056, 2005.

- 
- [118] "Sistema de control y supervisión en red basado en técnicas de Inteligencia Artificial para optimizar procesos de mecanizado a alta velocidad (COREMAV)," Ministerio de Educación y Ciencia. DPI2005-04298, 2006 - 2008.
- [119] "Sistema de control y supervisión en red para optimizar procesos de fresado a alta velocidad (COREFAV)," Ministerio de Educación y Ciencia. PROFIT. CIT-020400-2005-6, 2005 - 2007.
- [120] "Monitorización inteligente de los procesos de corte en la nano-escala en la fabricación de componentes aeronáuticos especiales (NANOCUT-INT)," Ministerio de Ciencia e Innovación. CIT420000-2008-13, 2008 - 2009.
- [121] "Investigación Industrial sobre Monitorización Inteligente y en Red de Procesos de Taladrado de Alto Rendimiento de Componentes Aeronáuticos Especiales (MONINTDRILL)," FUAM 082100, 2009 - 2010.
- [122] A. Isturiz, M. D. Riera, and J. M. Prado, "Experimental study of die filling in powder metallurgy," *Revista de Metalurgia*, vol. SPEC. VOL., pp. 181-186, 2005.
- [123] A. Ramírez, R. Morales, A. J. Morales, A. Ramos, and G. Solorio, "Development of industry processes simulators. Part III (Continuous casting)," *Revista de Metalurgia*, vol. 42, pp. 216-221, May-Jun 2006.
- [124] A. Ramírez, R. Morales, A. Ramos, and G. Solorio, "Development of industry processes simulators. Part I. (Continuous casting)," *Revista de Metalurgia*, vol. 42, pp. 203-208, May-Jun 2006.
- [125] A. Ramírez, A. Mosqueda, V. Sauce, R. Morales, A. Ramos, and G. Solorio, "Development of industry processes simulators. Part II (Continuous casting)," *Revista de Metalurgia*, vol. 42, pp. 209-215, May-Jun 2006.
- [126] M. A. Ramírez-Argaez, J. Tapia, J. Espinoza, and E. Alcantar, "Mathematical modelling of mixing in gas stirred ladles," *Revista de Metalurgia*, vol. 42, pp. 56-75, Jan-Feb 2006.
- [127] L. Ljung, *System identification: Theory for the user.*, 1st ed. New York: Prentice-Hall, Englewood Cliffs, 1987.
- [128] Y. Zhu, "Multivariable system identification for process control," Amsterdam: Pergamon, 2001, p. 349.
- [129] S. Chen, S. A. Billings, and W. Luo, "Orthogonal Least-Squares Methods and Their Application to Non-Linear System-Identification," *International Journal of Control*, vol. 50, pp. 1873-1896, Nov 1989.
- [130] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. New Jersey: Prentice Hall, 1999.
- [131] H. Kleines, J. Sarkadi, F. Suxdorf, and K. Zwill, "Measurement of real-time aspects of Simatic® PLC operation in the context of physics experiments," *IEEE Transactions on Nuclear Science*, vol. 51, pp. 489-494, 2004.
- [132] S. Vitturi, "Stochastic model of the profibus DP cycle time," *Iee Proceedings-Science Measurement and Technology*, vol. 151, pp. 335-342, Sep 2004.
-

- [133] M. Kupczyk and W. Misiak, "Tool life of the edges coated with the c-BN plus h-BN coatings with different structures during hard machinable steel machining," *Revista de Metalurgia*, vol. SPEC. VOL., pp. 483-487, 2005.
- [134] J. M. Sánchez-Sola, M. A. Sebastián, M. S. Carrilero, J. M. González, and M. Marcos, "Characterization of the built-up edge and the built-up layer in the machining process of AA7050 alloy," *Revista de Metalurgia*, vol. SPEC. VOL., pp. 365-368, 2005.
- [135] R. E. Haber, J. E. Jiménez, A. Jiménez, and J. López-Coronado, "Cutting force model for high speed machining process," *Revista de Metalurgia*, vol. 40, pp. 247-258, Jul-Aug 2004.
- [136] J. L. Cantero, J. M. Sanchez-Sola, M. M. Tardio, M. A. Sebastian, M. S. Carrilero, M. Marcos, and M. H. Miguelez, "Tool wear in the dry drilling of Ti-6Al-4V. A SEM/EDS study," *Revista de Metalurgia*, vol. SPEC. VOL., pp. 355-360, 2005.
- [137] M. Álvarez, M. S. Carrilero, J. E. Ares, J. M. González, and M. Marcos, "Influence of the cutting speed-feed relation on the surface quality of drilled carbon fiber samples," *Revista de Metalurgia*, vol. SPEC. VOL., pp. 361-365, 2005.
- [138] F. Cus, M. Milfelner, and J. Balic, "An intelligent system for monitoring and optimization of ball-end milling process," *Journal of Materials Processing Technology*, vol. 175, pp. 90-97, Jun 2006.
- [139] B. M. Imani, M. H. Sadeghi, and M. A. Elbestawi, "An improved process simulation system for ball-end milling of sculptured surfaces," *International Journal of Machine Tools & Manufacture*, vol. 38, pp. 1089-1107, Sep 1998.
- [140] S. Engin and Y. Altintas, "Mechanics and dynamics of general milling cutters. Part I: helical end mills," *International Journal of Machine Tools & Manufacture*, vol. 41, pp. 2195-2212, Dec 2001.
- [141] M. H. Sadeghi, H. Haghighat, and M. A. Elbestawi, "A solid modeler based ball-end milling process simulation," *International Journal of Advanced Manufacturing Technology*, vol. 22, pp. 775-785, Dec 2003.
- [142] E. E. M. Lim, H. Y. Feng, C. H. Menq, and Z. H. Lin, "The Prediction of Dimensional Error for Sculptured Surface Productions Using the Ball-End Milling Process .1. Chip Geometry Analysis and Cutting Force Prediction," *International Journal of Machine Tools & Manufacture*, vol. 35, pp. 1149-1169, Aug 1995.
- [143] Y. Altintas and P. Lee, "Mechanics and dynamics of ball end milling," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 120, pp. 684-691, 1998.
- [144] J. Thusty and F. Ismail, "Special Aspects of Chatter in Milling," *Journal of Vibration Acoustics Stress and Reliability in Design-Transactions of the ASME*, vol. 105, pp. 24-32, 1983.
- [145] Y. Altintas, *Manufacturing Automation*. Cambridge, United Kingdom: Cambridge University Press, 2000.
- [146] M. E. Martellotti, "An analysis of the milling process.," *Transactions of the ASME*, vol. 63, pp. 677-700, 1941.



- 
- [147] M. E. Martellotti, "An analysis of the milling process. Part II: down milling.," *Transactions of the ASME*, vol. 67, pp. 233–251, 1945.
- [148] F. Abrari, M. A. Elbestawi, and A. D. Spence, "On the dynamics of ball end milling: Modeling of cutting forces and stability analysis," *International Journal of Machine Tools & Manufacture*, vol. 38, pp. 215-237, Mar 1998.
- [149] X. W. Liu, K. Cheng, A. P. Longstaff, M. H. Widiyanto, and D. Ford, "Improved dynamic cutting force model in ball-end milling. Part I: theoretical modelling and experimental calibration," *International Journal of Advanced Manufacturing Technology*, vol. 26, pp. 457-465, Sep 2005.
- [150] R. F. Hamade and F. Ismail, "A case for aggressive drilling of aluminum," *Journal of Materials Processing Technology*, vol. 166, pp. 86-97, Jul 2005.
- [151] L. H. Wang, P. Orban, A. Cunningham, and S. Lang, "Remote real-time CNC machining for web-based manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 20, pp. 563-571, Dec 2004.
- [152] F. R. Biglari and X. D. Fang, "Real-Time Fuzzy-Logic Control for Maximizing the Tool Life of Small-Diameter Drills," *Fuzzy Sets and Systems*, vol. 72, pp. 91-101, May 1995.
- [153] M. Nouari, G. List, F. Girot, and D. Coupard, "Experimental analysis and optimisation of tool wear in dry machining of aluminium alloys," *Wear*, vol. 255, pp. 1359-1368, Aug-Sep 2003.
- [154] R. G. Landers, A. G. Ulsoy, and Y. H. Ma, "A comparison of model-based machining force control approaches," *International Journal of Machine Tools & Manufacture*, vol. 44, pp. 733-748, Jun 2004.
- [155] J. B. Kim, S. J. Lee, and Y. P. Park, "Stable and Efficient Drilling Process by Active Control of the Thrust Force," *Mechanical Systems and Signal Processing*, vol. 8, pp. 585-595, Sep 1994.
- [156] M. Elhachimi, S. Torbaty, and P. Joyot, "Mechanical modelling of high speed drilling. 1. Predicting torque and thrust," *International Journal of Machine Tools & Manufacture*, vol. 39, pp. 553-568, Apr 1999.
- [157] M. Elhachimi, S. Torbaty, and P. Joyot, "Mechanical modelling of high speed drilling. 2: Predicted and experimental results," *International Journal of Machine Tools & Manufacture*, vol. 39, pp. 569-581, Apr 1999.
- [158] Y. Sheng and M. Tomizuka, "Intelligent modeling of thrust force in drilling process," *Journal of Dynamic Systems Measurement and Control-Transactions of the Asme*, vol. 128, pp. 846-855, Dec 2006.
- [159] R. E. Haber and J. R. Alique, "Nonlinear internal model control using neural networks: an application for machining processes," *Neural Computing & Applications*, vol. 13, pp. 47-55, Apr 2004.
- [160] Y. M. Zhang, X. L. Lin, X. D. Wang, and G. Q. Cai, "The study on the dynamic characteristic of high speed machine tool and experiment validate," in *Advances in Materials Manufacturing Science and Technology*. vol. 471-472 Zurich-Uetikon: Trans Tech Publications Ltd, 2004, pp. 765-769.
-

- [161] Y. T. Oh, W. T. Kwon, and C. N. Chu, "Drilling torque control using spindle motor current and its effect on tool wear," *International Journal of Advanced Manufacturing Technology*, vol. 24, pp. 327-334, 2004.
- [162] R. J. Furness, T. C. Tsao, J. S. Rankin, M. J. Muth, and K. W. Manes, "Torque control for a form tool drilling operation," *Ieee Transactions on Control Systems Technology*, vol. 7, pp. 22-30, Jan 1999.
- [163] Y. T. Oh, G. D. Kim, and C. N. Chu, "Design of a drilling torque controller for a machining center," *International Journal of Advanced Manufacturing Technology*, vol. 22, pp. 329-335, Oct 2003.
- [164] M. Pirtini and I. Lazoglu, "Forces and hole quality in drilling," *International Journal of Machine Tools & Manufacture*, vol. 45, pp. 1271-1281, Sep 2005.
- [165] J. Nilsson, B. Bernhardsson, and B. Wittenmark, "Stochastic analysis and control of real-time systems with random time delays," *Automatica*, vol. 34, pp. 1154-1154, Sep 1998.
- [166] L. A. Montestruque and P. Antsaklis, "Stability of model-based networked control systems with time-varying transmission times," *Ieee Transactions on Automatic Control*, vol. 49, pp. 1562-1572, Sep 2004.
- [167] J. R. Dormand and P. J. Prince, "A family of embedded Runge-Kutta formulae," *Journal of Computational and Applied Mathematics*, vol. 6, pp. 19-26, 1980.
- [168] R. M. Del Toro-Matamoros, R. E. Haber, J. Pamies, and A. Bustillo, "Modelling and simulation of the high-speed milling process based on MATLAB-SIMULINK," *Revista de Metalurgia*, vol. 44, pp. 176-188, 2008.
- [169] R. M. Del Toro, M. C. Schmittiel, R. E. Haber-Guerra, and R. Haber-Haber, "System identification of the high performance drilling process for network-based control," in *2007 Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC2007*, Las Vegas, NV, 2008, pp. 827-834.
- [170] K. J. Åström and T. Hagglund, *PID controllers : theory, design, and tuning*: Research Triangle Park, NC : Instrument Society of America, 1995.
- [171] A. Visioli, "Optimal tuning of PID controllers for integral and unstable processes," *IEE Proceedings: Control Theory and Applications*, vol. 148, pp. 180-184, 2001.
- [172] B. Russell, "Vagueness," *Australasian Journal of Philosophy and Psychology*, vol. 1, pp. 84-92, 1923.
- [173] L. A. Zadeh, "OUTLINE OF A NEW APPROACH TO THE ANALYSIS OF COMPLEX SYSTEMS AND DECISION PROCESSES," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, pp. 28-44, 1973.
- [174] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.
- [175] R. M. Tong, "A control engineering review of fuzzy systems," *Automatica*, vol. 13, pp. 559-569, 1977.
- [176] M. Braae and D. A. Rutherford, "Theoretical and linguistic aspects of the fuzzy logic controller," *Automatica*, vol. 15, pp. 553-577, 1979.

- 
- [177] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller - Part I," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 404-418, 1990.
- [178] C. Elkan, "Paradoxical success of fuzzy logic," in *Proceedings of the National Conference on Artificial Intelligence*, Washington, DC, USA, 1993, pp. 698-703.
- [179] P. Bernhard, "Fuzzy Control: Facts, Japan and Europe," *TECNews*, pp. 4-5, June 1992.
- [180] A. Kaufmann, "Introduction to the Theory of Fuzzy Subsets.," New York: Academic Press, 1975.
- [181] H. Zimmermann, "Fuzzy Sets Theory and Its Applications," Kluwer Academic Pub., 1985.
- [182] D. Dubois and H. Prade, "Fuzzy sets, probability and measurement," *European Journal of Operational Research*, vol. 40, pp. 135-154, 1989.
- [183] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller - Part II," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 419-435, 1990.
- [184] L. A. Zadeh, "Similarity relations and fuzzy orderings," *Information Sciences*, vol. 3, pp. 177-200, 1971.
- [185] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-I," *Information Sciences*, vol. 8, pp. 199-249, 1975.
- [186] R. E. Haber Guerra, "Control Jerárquico Inteligente. Aplicación al Proceso de Fresado," in *Dpto. de Automática, Ingeniería Electrónica e Informática Industrial*. vol. Ph D. Madrid: Universidad Politécnica de Madrid, E. T. S. de Ingenieros Industriales, 1998.
- [187] F. Matía and A. Jiménez, "On optimal implementation of fuzzy controllers," *Int J Intell Contr Syst*, vol. 1, pp. 407-415, 1996.
- [188] R. M. Tong, "The construction and evaluation of fuzzy models," *Advances in Fuzzy Set Theory and Applications*, pp. 559-576, 1979.
- [189] C. A. Cabezas and R. Haber, "Objetivación de los algoritmos de control borrosos," in *IV Congreso Latinoamericano de Control Automático* Puebla, México, 1990.
- [190] C. Rodríguez, R. Haber, and J. R. Alique, "Modeling of Machine-Tool Cutting Process Applying Fuzzy Logic and Clustering Analysis," in *II European Congress on Intelligent Techniques and Soft Computing, EUFIT'94*. vol. II Aachen, Germany, 1994, pp. 1071-1075.
- [191] T. Takagi and M. Sugeno, "DERIVATION OF FUZZY CONTROL RULES FROM HUMAN OPERATOR'S CONTROL ACTIONS," in *IFAC Proceedings Series*, Marseille, Fr, 1984, pp. 55-60.
- [192] W. Pedrycz, *Fuzzy Control and Fuzzy Systems*. New York: John Wiley & Sons, Inc., 1989.
- [193] P. J. King and E. H. Mamdani, "The application of fuzzy control systems to industrial processes," *Automatica*, vol. 13, pp. 235-242, 1977.
- [194] M. Braae and D. A. Rutherford, "Selection of parameters for a fuzzy logic controller," *Fuzzy Sets and Systems*, vol. 2, pp. 185-199, 1979.
-

- [195] R. M. Tong, "FUZZY CONTROL SYSTEMS: A RETROSPECTIVE," in *Proceedings of the American Control Conference*, San Francisco, CA, USA, 1983, pp. 1224-1229.
- [196] M. Mizumoto and H. J. Zimmermann, "Comparison of fuzzy reasoning methods," *Fuzzy Sets and Systems*, vol. 8, pp. 253-283, 1982.
- [197] L. I. Larkin, "A fuzzy logic controller for aircraft flight control," in *Industrial Applications of Fuzzy Control*, 1985, pp. 87-104.
- [198] J. M. Keller and D. J. Hunt, "INCORPORATING FUZZY MEMBERSHIP FUNCTIONS INTO THE PERCEPTRON ALGORITHM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, pp. 693-699, 1985.
- [199] S. Mitra and S. K. Pal, "Fuzzy multi-layer perceptron, inferencing and rule generation," *IEEE Transactions on Neural Networks*, vol. 6, pp. 51-63, 1995.
- [200] J. Kim and N. Kasabov, "HyFIS: Adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems," *Neural Networks*, vol. 12, pp. 1301-1319, 1999.
- [201] D. Nauck, F. Klawonn, and R. Kruse, *Foundations of Neuro-Fuzzy Systems*: John Wiley & Sons, Inc., 1997.
- [202] L. X. Wang, *A course in fuzzy systems and control*: Prentice-Hall, Inc., 1997.
- [203] N. K. Kasabov, J. Kim, M. J. Watts, and A. R. Gray, "FuNN/2 - A fuzzy neural network architecture for adaptive learning and knowledge acquisition," *Information Sciences*, vol. 101, pp. 155-175, 1997.
- [204] N. K. Kasabov, "Learning fuzzy rules and approximate reasoning in fuzzy neural networks and hybrid systems," *Fuzzy Sets and Systems*, vol. 82, pp. 135-149, 1996.
- [205] N. K. Kasabov and Q. Song, "DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction," *IEEE Transactions on Fuzzy Systems*, vol. 10, pp. 144-154, 2002.
- [206] Q. Song and N. Kasabov, "TWNFI - a transductive neuro-fuzzy inference system with weighted data normalization for personalized modeling," *Neural Networks*, vol. 19, pp. 1591-1596, 2006.
- [207] R. Babuška and H. B. Verbruggen, "An overview of fuzzy modeling for control," *Control Engineering Practice*, vol. 4, pp. 1593-1606, 1996.
- [208] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*. New Jersey, USA: Prentice-Hall, Inc., 1997.
- [209] P. K. Dash, S. K. Panda, T. H. Lee, J. X. Xu, and A. Routray, "Fuzzy and neural controllers for dynamic systems: An overview," in *Proceedings of the International Conference on Power Electronics and Drive Systems*, Singapore, 1997, pp. 810-816.
- [210] S. S. Rodil and M. J. Fuente, "Fault-tolerant nonlinear IMC control," *Control IMC no lineal tolerante a fallos*, vol. 4, 2007.
- [211] J. Vieira, F. M. Dias, and A. Mota, "Artificial neural networks and neuro-fuzzy systems for modelling and controlling real systems: A comparative study," *Engineering Applications of Artificial Intelligence*, vol. 17, pp. 265-273, 2004.

- 
- [212] S. Altug, M. Y. Chow, and H. Joel Trussell, "Fuzzy inference systems implemented on neural architectures for motor fault detection and diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 46, pp. 1069-1079, 1999.
  - [213] L. C. Chang and F. J. Chang, "Intelligent control for modelling or real-time reservoir operation," *Hydrological Processes*, vol. 15, pp. 1621-1634, 2001.
  - [214] S. Singer and J. Nelder, "Nelder-Mead algorithm," *Scholarpedia*, vol. 4(7), 2009.
  - [215] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671-680, 1983.
  - [216] J. H. Holland, *Adaptation in natural and artificial system: an introduction with application to biology, control and artificial intelligence*: Ann Arbor, University of Michigan Press, 1975.
  - [217] J. J. Grefenstette, " Genetic Algorithms and Their Applications," in *Second International Conference on Genetic Algorithms*, Cambridge, MA, 1987.
  - [218] P. T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, pp. 19-67, 2005.
  - [219] R. Rubinstein and D. Kroese, *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Berlin, Heidelberg, New York: Springer, 2004.
  - [220] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods," *SIAM Review*, vol. 45, pp. 385-482, 2003.
  - [221] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions," *SIAM Journal on Optimization*, vol. 9, pp. 112-147, 1999.
  - [222] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, pp. 1087-1092, 1953.
  - [223] W. K. Hastings, "Monte carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, pp. 97-109, 1970.
  - [224] L. Ingber, "Adaptive simulated annealing (ASA): Lessons learned," in *Polish Journal Control and Cybernetics*, 1995.
  - [225] J. d. D. Sanz Bobi, "Control de procesos basado en eventos mediante bases de conocimiento borrosas y algoritmos genéticos," in *Automática, Ingeniería Electrónica e Informática Industrial* Madrid: Universidad Politécnica de Madrid, 2001.
  - [226] R. C. Fernández, J. M. Luque, M. L. Gallego, A. T. González, and T. G. Nuñez, "Algoritmos genéticos para la resolución de problemas de Programación por Metas Entera. Aplicación a la Economía de la Educación.," in *X JORNADAS MADRID 2002*. vol. Actas\_10: Rect@: Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA, 2002, p. 42.
  - [227] D. E. Goldberg, "A comparison of selection schemes used in genetic algorithm," in *Foundations of Genetic Algorithms*, G. E. Rawlins, Ed.: Morgan Kaufmann Publishers, 1991, pp. 69-93.
-

- [228] J. H. Holland, K. J. Holyoak, R. E. Nisbett, and P. R. Thagard, *Induction: Processes of inference, learning, and discovery.*: Cambridge, MA: MIT Press, 1986.
- [229] J. J. Grefenstette, "Genetic Algorithms and Their Applications," in *First International Conference on Genetic Algorithms*, Pittsburgh, PA, 1985.
- [230] R. Rubinstein, "A stochastic minimum cross-entropy method for combinatorial optimization and rare-event estimation," *Methodology and Computing in Applied Probability*, vol. 7, pp. 5-50, 2005.
- [231] D. P. Kroese, R. Y. Rubinstein, and T. Taimre, "Application of the cross-entropy method to clustering and vector quantization," *Journal of Global Optimization*, vol. 37, pp. 137-157, 2007.
- [232] R. Haber-Haber, R. Haber, M. Schmittziel, and R. M. del Toro, "A classic solution for the control of a high-performance drilling process," *International Journal of Machine Tools and Manufacture*, vol. 47, pp. 2290-2297, 2007.
- [233] K. J. Åström and T. Hägglund, "PID controllers: theory, design, and tuning," 3th ed: International Society for Measurement and Control, 1998.
- [234] A. Gajate, R. E. Haber, and R. M. del Toro, "Neurofuzzy Drilling Force-Based Control in an Ethernet-Based Application," *International Journal of Innovative Computing, Information and Control*, vol. 6, pp. 373-386, January 2010.
- [235] R. Carotenuto, "An iterative system inversion technique," *International Journal of Adaptive Control and Signal Processing*, vol. 15, pp. 85-91, 2001.
- [236] R. E. Haber-Guerra, R. Haber-Haber, D. M. Andrés, and A. A. Palomar, "Networked fuzzy control system for a high-performance drilling process," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 130, pp. 0310091-0310096, 2008.
- [237] F. L. Lewis and V. L. Syrmos, *Optimal control*. New York: Wiley-Interscience, 1995.
- [238] R. Middleton and G. Goodwin, "Digital Control and Estimation: A unified approach," New York, 1990.
- [239] R. R. Yager and D. Filev, "Essentials of Fuzzy Modeling and Control," *John Wiley&Sons Inc.*, 1994.
- [240] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy controller," *International Journal of Man-Machine Studies*, vol. 7, pp. 1-13, 1975.
- [241] P. J. MacVicar-Whelan, "Fuzzy sets for a man-machine interactions," *International Journal of Man-Machine Studies*, vol. 8, pp. 687-697, 1977.
- [242] K. C. Lee, S. Lee, and M. H. Lee, "Remote fuzzy logic control of networked control system via profibus-DP," *IEEE Transactions on Industrial Electronics*, vol. 50, pp. 784-792, 2003.
- [243] O. Cerdón, F. Herrera, and A. Peregrín, "Applicability of the fuzzy operators in the design of fuzzy logic controllers," *Fuzzy Sets and Systems*, vol. 86, pp. 15-41, 1997.
- [244] S. Roychowdhury and W. Pedrycz, "A survey of defuzzification strategies," *Int. Journal of Intelligent Systems*, vol. 16, pp. 679-695, 2001.

- [245] T. C. Yang, "Networked control system: a brief survey," *IEE Proceedings-Control Theory and Applications*, vol. 153, pp. 403-412, Jul 2006.
- [246] A. Sala, T. M. Guerra, and R. Babuška, "Perspectives of fuzzy systems and control," *Fuzzy Sets and Systems*, vol. 156, pp. 432-444, 2005.
- [247] J. M. A. Márquez, A. J. B. Piña, and M. E. G. Arias, "A general and formal methodology to design stable nonlinear fuzzy control systems," *IEEE Transactions on Fuzzy Systems*, vol. 17, pp. 1081-1091, 2009.
- [248] R. E. Precup, S. Doboli, and S. Preitl, "Stability analysis and development of a class of fuzzy control systems," *Engineering Applications of Artificial Intelligence*, vol. 13, pp. 237-247, 2000.
- [249] D. Wang and C. B. Soh, "On regularizing singular systems by decentralized output feedback," *IEEE Transactions on Automatic Control*, vol. 44, pp. 148-152, 1999.
- [250] Y. Zhao and E. G. Collins Jr, "Fuzzy PI control design for an industrial weigh belt feeder," *IEEE Transactions on Fuzzy Systems*, vol. 11, pp. 311-319, 2003.
- [251] B. M. Mohan and A. Sinha, "Analytical structure and stability analysis of a fuzzy PID controller," *Applied Soft Computing Journal*, vol. 8, pp. 749-758, 2008.
- [252] R. E. Haber, G. Schmitt-Braess, R. H. Haber, A. Alique, and J. R. Alique, "Using circle criteria for verifying asymptotic stability in PI-like fuzzy control systems: application to the milling process," *IEE Proceedings-Control Theory and Applications*, vol. 150, pp. 619-627, Nov 2003.
- [253] D. Y. Sun and P. M. Lin, "The solution of time optimal control problems by simulated annealing," *Journal of Chemical Engineering of Japan*, vol. 39, pp. 753-766, 2006.
- [254] R. E. Haber, R. M. del Toro, and A. Gajate, "Optimal fuzzy control system using the cross-entropy method. A case study of a drilling process," *Information Sciences*, 2010.